AD-A068 874 NAVAL UNDERSEA WARFARE CENTER SAN DIEGO CA F/G 17/1
ADAPTATION OF COMPUTER PROGRAMS FOR THE DIGITAL SIMULATION OF T--ETC(U)
JUL 68 J MUNOZ-FLORES, S K BUEHLER

UNCLASSIFIED NUWC-TN-149 NL

1 OF 2

AD
A068874

UNCLASSIFIED - MOST Project -1

(14) NUWC-TN-149

[redacted]
~~GROUP 4~~
~~Downgraded at 3-year intervals;~~
~~Declassified after 12 years.~~

LEVEL II

(1)

B.S.

001992

AD A068874

NAVAL UNDERSEA WARFARE CENTER

(11) JULY 1968

(12) 135p.

(9) Technical note,

(6)

ADAPTATION OF COMPUTER PROGRAMS FOR THE DIGITAL SIMULATION OF THE WAVE PERIOD PROCESSOR (U)

(10) by J. Munoz-Flores ~~and~~ S. K. Buehler

San Diego, California

(16) F11121 (17) SF11121100

SUBPROJECT NO. SF11-121-100

TASK NO. 11197

149

00V1C

1cy

403 023

680628-0073

[redacted]

17

001992

118306

UNCLASSIFIED

PREFACE

This technical note describes an adapted digital simulation program for the wave period processor. It was written by Code D603, Naval Undersea Warfare Center, San Diego Division, Subproject SF 11 121 100, Task 11197 (NUWC Problem E1-19). This document has been prepared because it is believed the information contained herein will be useful to others at the Naval Undersea Warfare Center (NUWC) and to a few persons outside NUWC.

TABLE OF CONTENTS

I. INTRODUCTION

The purpose of this technical note is to document the adaptation of a Wave Period Processor (WPP) Digital Simulation Program to NUWC-SD facilities.

The Sperry Gyroscope Company of Long Island, New York developed a digital simulation program of the Wave Period Processor "to confirm the performance characteristics relating to the incentive test measurements". This simulation program was made available in December 1967 to the PAIR Project Office, Code D554.

Code D603, in its quest to enhance in-house capabilities relating to digital simulation of signal processors, undertook to adapt and implement the Sperry Programs on the AN/USQ-20 computers. The adaptation of these programs to NUWC facilities will result in the following benefits:

a. PAIR's system analysis group will be able to analyze and predict systems performance in a timely manner.

b. In-house capability will be enhanced by exposure to the approaches being used by others who are active in the field.

II. PROGRAM, GENERAL

The block diagram Wave Period Processor digital simulation is shown in Figure 1.

The major task in the adaptation of this simulation program consisted of:

a. Generation of a new random number generator, and

b. Rewriting of the routines for implementation on the AN/USQ-20 computers.

[redacted]

The Wave Period Processor program is modularized into 3 main independent subroutines.

a. NOISE subroutine

b. WAVE subroutine

c. PROCS subroutine

The above subroutines perform the following functions:

a. The NOISE Subroutine .

1. Generates random gaussian-distributed noise.

2. Correlates the random noise.

3. Interpolates and centers the noise at 20 kHz for the resulting narrow band gaussian noise.

b. The WAVE Subroutine

1. Combines narrow band gaussian noise and linear FM signal according to specified S/N ratios.

2. Measures each twelve period interval of the wave.

3. Computes the modulo-32 integer values of the wave measurements.

c. The PROCS Subroutine

1. Generates the zone logic

2. Performs range bin detection

3. Computes statistical information

In order to better understand the functions of the WPP Simulation Programs as related to the hardware, a brief description of the hardware system will be presented.

III. WAVE PERIOD PROCESSOR DESCRIPTION*

The WPP is an active search signal processor. It receives signals continuously from all 48 active beams and outputs digital amplitude as a function of bearing and time (range). The output data appears on the B-Scan CRT display as horizontal strokes in a matrix of 48 bearing bins by 240 range bins. Strokes may depict returns from a single transmitted ping or from five pings, in which case the 240 range bins are reduced to 48, each of which accommodates 5 strokes representing the same range.

The 48 beams are amplified and heterodyned to 20 kHz at which frequency very narrow bandpass filters are available (about 440 Hz). All 48 channels are then clipped and passed through "digitizers". A "digitizer" is a frequency measuring device which expresses the frequency of the input waves as an output 5-bit number. With the 440 Hz input bandwidth divided into 32 intervals, each of the possible 32 output numbers represents a frequency interval of some 14 Hz. Forty-eight zone logic and integrator circuits examine the modulo-32 numbers from the digitizers for a pattern of counts which reveals the presence of the linear FM slide in the received signal. Logic zones are established, and input signal counts are made in these zones. Noise or dissimilar input signals produce equal counts in all zones; deviations are indicative of signal returns, the magnitudes representing the signal strength. "Integration and peak detection", as used here, refer to a process of accumulating samples and storing the largest counts for

*PAIR System Docket 9287-02120, Sperry Rand Corporation outputting serially through Display Control to the B-Scan CRT. A count is obtained on each of the 48 bearing channels every 720 microseconds which is every 15 microseconds per datum.

IV. NOISE SUBROUTINE

A. Noise, General

As the first stage of the Wave Period Processor simulation, the noise simulation is the standard representation of bandpass noise as the sum of uncorrelated quadrature components. The random (pseudo-random) noise generator provides two independent samples and this uncorrelated gaussian noise is passed into identical digital filters synthesized from the impulse response of the low pass equivalent, 470 Hz bandpass filter in the system. The outputs from the filters are the amplitudes of the in-phase and quadrature components of the correlated noise. These outputs are respectively multiplied by the sine and cosine of the carrier frequency to obtain the in-phase and quadrature components of the correlated narrow band noise at bandpass. The components are then summed and recorded on magnetic tape. Constants for the digital filter used to simulate the 470 Hz filter are generated by another program.

B. Noise, Detailed

The noise program KAVEE and its subroutines BLOCK, GAUSS, and RANDM generate approximately 16 seconds of simulated real-time random noise. Each of the 101 blocks contains 16,128 independent samples, each sample or word representing 10 microseconds of data. This block size corresponds to the requirement of the wave period program in which the signal wave repeat cycle is 0.16128 seconds.

The random number generator RANDM* produces a pseudo-random number uniformly distributed between zero and one. The routine GAUSS* obtains two random numbers from RANDM, R_1 and R_2, and in turn delivers $G_1 = (-2 \ln R_1)^{1/2} \sin 2\pi R_2$ and $G_2 = (-2 \ln R_1)^{1/2} \cos 2\pi R_2$ which are independently gaussian distributed with zero mean and unit variance.

The subroutine BLOCK performs the major computations for one block of noise (16128 samples). Input parameters to the noise program for sampling time (T), Bandwidth (BW), and the Z-transform coefficients (C1-C9) of the Butterworth filter are computed elsewhere by the program CONSTS (plus routines BUTRW5 and INVRT6) and are read from cards by BLOCK. These are read only once, since the same values are used for each of the 101 blocks of noise which comprise one complete noise run. For the first runs, T=50 microseconds, BW=471.2 Hz.

The two gaussian random numbers represent uncorrelated noise samples for the sine and cosine channels with zero mean and unit variance. These are produced every 50 microseconds. The uncorrelated samples are passed through the two-step recurrence equations to obtain low-pass correlated noise.

For Sine Channel:

$$YS_i = C_6 \cdot XS_{i-1} + C_7 \cdot XS_{i-2} + C_8 \cdot XS_{i-3} + C_9 \cdot XS_{i-4}$$

where:

$$XS_i = C_1 \cdot XS_{i-1} - C_2XS_{i-2} + C_3 \cdot XS_{i-3} - C_4 \cdot XS_{i-4} + C_5 \cdot XS_{i-5} + G_1$$

*Complete descriptions of these routines are enclosed in Appendix A

For Cosine Channel:

$$YC_i = C_6 \cdot XC_{i-1} + C_7 \cdot XC_{i-2} + C_8 \cdot XC_{i-3} + C_9 \cdot XC_{i-4}$$

where:

$$XC_i = C_1 \cdot XC_{i-1} - C_2 \cdot XC_{i-2} + C_3 \cdot XC_{i-3} - C_4 \cdot XC_{i-4} + C_5 \cdot XC_{i-5} + G_2$$

C1-C9 are the Z-transform coefficients of the Butterworth filter. G_i is the zero mean, unit variance random number generated by subroutine GAUSS.

A three-point Lagrange interpolation is applied to the low-pass outputs of the second stage recurrence equations to produce one sample every 10 microseconds. Narrow band noise is desired, centered at 20 kHz. The low-pass outputs are therefore multiplied respectively by the sines and cosines of multiples of 72°, which correspond to 10 microseconds of a 20 kHz wave, and the two channels are combined.

For $w1 = \sin \omega_o t_i$ and $w2 = \cos \omega_o t_i$,

$$Y_i = L_1 \cdot w1 \cdot YS_{i-2} + L_2 \cdot w1 \cdot YS_{i-1} + L_3 \cdot w1 \cdot YS_i + L_1 \cdot w2 \cdot YC_{i-2}$$

$$+ L_2 \cdot w2 \cdot YC_{i-1} + L_3 \cdot w2 \cdot YC_i$$

where L_i is the Lagrange interpolation constant, YS_i's and YC_i's are the low-pass outputs from the sine and cosine channel second stage recurrence equations, ω_o is the desired channel band center, and Y_i is the final output value.

In actuality, the constants $L_1 w1$ through $L_3 w2$ in the above equation are pre-multiplied. The L_i's are the three-point interpolation constants for the intervals -0.4, -0.2, 0.0, +0.2, +0.4.* The w_i's are the numerical values for the sines and cosines of 72°, 36°, and 0°.

*p.70, Lagrangian Methods, Introduction to Numerical Analysis, F.B. Hildebrand

Five noise samples, Y_i, are therefore generated each cycle of the recurrence and interpolation/centering equations, one each 10 microseconds for the original 50 microsecond sample. However, since 5 is an uneven factor of 16128, the noise block size, a modulo- 5 indexing scheme has been employed. In a set of 5 blocks:

J = 0: Block 1 - 16130 samples are computed.

J = 2: Block 2 - 2 extra samples from first block begin second block, and 16130 new samples are computed.

J = 4: Block 3 - 4 extra samples from second block begin third block, and 16125 new samples are computed.

J = 1: Block 4 - 1 extra sample from third block begins fourth block, and 16130 new samples are computed.

J = 3: Block 5 - 3 extra samples from fourth block begin fifth block, and 16125 new samples are computed.

The extra samples computed for one block of noise are moved to the beginning of the next block and the noise function is therefore continuous. A maximum of 16132 samples may be present at one time, even though only 16128 samples are used in any one noise block for output.

The subroutine BLOCK concludes by writing one block of 16128 narrow band gaussian noise values on magnetic tape and computing the mean of the sum and the mean of the sum of the squares for the 16128 noise values.

The control routine for the noise generation, KAVEE, calls on the subroutine BLOCK a total of 101 times for one complete noise run. Each time the control routine calls the subroutine BLOCK, plus once after the final call, it will print the BLOCK input parameters, including the current block number, indices, modulo- 5 index, initial entry flag, current starting point of the random number generator, and current values of the two step recurrence equations. The control routine will output the mean and mean of squares values and compute and output the variance and standard deviation, all for each noise block. At the end of a complete run, it will also compute and output the mean, mean of squares, variance, and standard deviation from the mean values of all noise blocks combined.

The total output for 101 blocks of noise is approximately 1500 feet of magnetic tape. Since each group of 16,128 words is needed to generate 224 samples of the wave period measurements, one complete noise tape will provide 101 x 224 = 22,624 samples of wave period measurement.

Symbol Correspondence

ABS Library routine for absolute value of a real argument.

ALOG Library routine for natural logarithm of a real argument.

ANGLE Second of the pair of random numbers used to compute two gaussian distributed numbers.

BLKS Maximum number of noise blocks, in floating point.*

BLOCK Name of subroutine which performs the noise computations.

BW Bandwidth of Butterworth filter (471.2 Hz).**

*Mixed mode arithmetic is not available; therefore, constants must be entered in any and all modes in which they are used.
**Numbers in parentheses refer to values used in first NUWC run.

C1-C9 Z-transform coefficients of Butterworth filter.

CA1-CA3 Three point Lagrange interpolation constants for the

CB1-CB3 cosine channel.

COS Library routine for trigonometric cosine of a real argument.

DENOM Number of samples for one noise block, in floating point.

DUMMY Dummy argument used to set the random number generator.

G Array containing 1/MULTP of a noise block (288).
(previously* equal to length of one magnetic tape record)

GAUSS Function subroutine which computes gaussian distributed random numbers.

II Indicator for number of noise blocks written on output tape.

IFLAG Indicator for first or second entry to routine GAUSS
IFLAG = 0, pick up sine channel gaussian value
IFLAG = 1, pick up cosine channel gaussian value (no longer used in routine BLOCK)

IMAX Major processing index for the 5-step noise generation equations.

IRAND3 Entry to random number generator that will obtain the last random number, which is also the starting point for the next random number.

J Modulo 5 counter for shifting extra values (> 16128) from the end of one noise block to the beginning of the next block.

:

*Refers to Sperry program format.

KAVEE Main control routine for noise generation.

LNGTH Intermediate number of noise points generated equalling 1/MULTP of one block of noise (288) (originally equal to number of noise samples on one magnetic tape record.*)

MLTM1 Equal to MULTP-1 for indexing purposes (55).

MOD Library routine for remaindering integer arguments.

MULTP Multiplication factor which, along with LNGTH, determines the length of 1 block of noise (56) (previously equal to the number of magnetic tape records for one noise block.*)

NBLKS Maximum number of noise blocks.

NBLOCK Index and indicator for current noise block being processed.

NTRY Flag which indicates initial pass in a given execution of program.

= 0, initial pass

= 1, succeeding pass

NUMBR Number of noise samples in one noise block

= MULTP x LNGTH

RANDM1 Entry to random number generator to obtain the next pseudo-random number.

RANDM2 Entry to random number generator to store a new starting point for generator.

RNDSET Octal number used to start the random number generator at a particular point.

*Refers to Sperry program format.

CONFIDENTIAL CONFIDENTIAL

MLTM1 Equal to MULTP-1 for indexing purposes (55)

MOD Library routine for remaindering integer arguments

SA1-SA3 SB1-SB3 Three point Lagrange interpolation constants for the sine channel

SIGY Standard deviation for one block of noise.

SIGZ Standard deviation for all blocks of noise combined.

SIN Library routine for trigonometric sine of a real argument.

SQRT Library routine for square root of a real argument.

SSQY Sum of the squares of samples for one block of noise.

SSQZ Sum of the squares of samples for all blocks of noise combined.

SUMY Sum of samples for one block of noise.

SUMZ Sum of samples for all blocks of noise combined.

T Butterworth filter sampling time.

TEMP Temporary location for intermediate calculations.

TRANR First of the pair of random numbers used to compute two gaussian distributed numbers.

VARY Variance of samples for one block of noise.

VARZ Variance of samples for all blocks of noise combined.

XC Array containing the previous four and the current first-step recurrence noise numbers for the cosine channel.

XS Array containing the previous four and the current first-step recurrence noise numbers for the sine channel.

Y Array containing noise samples.

YBAR Mean of samples for one block of noise.

YC Array containing the previous two and the current second-step recurrence noise numbers for the cosine channel.

YMSQ Mean of squares of samples for one block of noise.

YS Array containing the previous two and the current second-step recurrence noise numbers for the sine channel.

ZBAR Mean of sample for all noise blocks combined.

ZMSQ Mean of squares of samples for all noise blocks combined.

INPUT

4 punched cards contain input values for the program:

Card	Variable	Columns	Format*
1	T	5-27	E23.16
	BW	32-54	E23.16
2	C_1	4-26	E23.16
	C_2	31-53	E23.16
	C_3	58-80	E23.16
3	C_4	5-27	E23.16
	C_5	32-54	E23.16
4	C_6	5-19	E15.8
	C_7	24-38	E15.8
	C_8	43-57	E15.8
	C_9	62-76	E15.8

In the case of the noise program, double precision variables T, BW, C1-C5 are entered through an E field.

OUTPUT

A list of the Z-transform coefficients, the sampling time, and bandwidth of the Butterworth filter are printed out as soon as they are read in from the cards.

At the beginning of each noise block computation, the noise block number, two indices which combined indicate the length of a noise block, the modular 5 index, initial pass flag, and the last number (in octal integer form) generated by the random number routine are printed.

*"In format statements used for data input, E, F, and G field specifications are interchangeable. In addition, Double Precision variables may be entered with those specifications". Fortran IV for the NUWC (SD) 1230 and AN/USQ-20 computers.

The last random number is useful if one wishes to stop the program and restart it later at the point the random number generator left off. The call, RANDM2 (RNDSET), to the random number generator will put the variable RNDSET in as the starting point.

The sine and cosine recurrence tables XS, XC, YS, YC are printed out so that they may also be entered to restart the program.*

At the end of each noise block computation, the mean of the sum, mean of the sum of squares, variance, and standard deviation of the 16128 samples are printed out.

After one entire noise run (101 blocks) has been completed, the mean of the sum, mean of the sum of squares, variance, and standard deviation are printed for all noise blocks combined.

The 101 noise blocks of 16128 samples each are written identically on two IBM compatible magnetic tapes, and an end of file mark completes the output on each tape.

Subroutines

Descriptions of functions and entry points of the routines GAUSS and RANDM follow.

Library functions used include:

MOD Computes remainder of two integer arguments.

SQRT Computes square root of a real argument.

plus those called by the routine GAUSS

SQRT (Above)

ABS Computes absolute value of a real argument.

* See "Consecutive Runs of Noise Program"

ALOG Computes natural logarithm of a real argument.

SIN Computes trigonometric sine of a real argument.

COS Computes trigonometric cosine of a real argument.

Execution

Fortran IV on the NUWC (SD) 1230 and AN/USQ-20 computers is an automatic compile and execute system, so the entire program deck, as listed following, is used for a run. A PAUSE has been inserted to stop the program at the beginning of execution for the purpose of mounting the output tapes.

Tape unit assignments for NUWC-SD Univac 1230 or AN/USQ-20 computers:

System tape on Unit M1

Noise output tape on Unit M2

Noise output tape on Unit M3

Execution Time

To generate one block of noise data of 16128 samples takes 25 minutes on the Univac AN/USQ-20 computer and 1 minute 15 seconds on the Univac 1230 computer. However, in order to take advantage of the fast double precision hardware on the 1230 computer, the entire program was converted to double precision, with a running time of 50 seconds per noise block.

Accuracy

Double precision arithmetic is used in the recurrence formulas to generate the XS_i and XC_i values. Single-precision arithmetic is used to generate the noise values for output on magnetic tape. However, as stated above, the entire program was also written in double precision to take advantage of the fast floating point hardware on the Univac 1230 computer. The program in both forms is included in Appendix C.

Consecutive Runs of Noise Program

When more than 101 blocks of continuous noise are desired, the last values from the two-step recurrence equation tables and the last starting point of the random number generator must be initially entered into the noise program. Seven cards contain the generator starting point, RNDSET, the five values for the first-step recurrence equation for both the sine (XS) and cosine (XC) channels, and the three values for the second-step recurrence equation for the sine (YS) and cosine (YC) channels. The double precision noise program included in this document contains the instructions for loading the above values.

CARD	VARIABLE	COLUMNS	FORMAT
1	RNDSET	9 - 20	012
2	XS(1)	4 - 26	D23.16
	XS(2)	30 - 52	D23.16
	XS(3)	56 - 78	D23.16
3	XS(4)	4 - 26	D23.16
	XS(5)	30 - 52	D23.16
4	XC(1)	4 - 26	D23.16
	XC(2)	30 - 52	D23.16
	XC(3)	56 - 78	D23.16
5	XC(4)	4 - 26	D23.16
	XC(5)	30 - 52	D23.16
6	YS(1)	4 - 26	D23.8*
	YS(2)	30 - 52	D23.8
	YS(3)	56 - 78	D23.8
7	YC(1)	4 - 26	D23.8
	YC(2)	30 - 52	D23.8
	YC(3)	56 - 78	D23.8

*The second-stage values would be entered through E w.d field for the primarily single precision noise program.

V. WAVE SUBROUTINE

A. Wave, General

The WAVE program is the second stage of the simulation, using as input the output, on magnetic tape, from the NOISE program. The noise samples are combined with a linear FM slide which is scaled to provide the proper input signal-to-noise ratio. The combined signal plus noise samples are then observed for a sign change from minus to plus. To increase the accuracy of the wave period measurement, an interpolation is made for the first and thirteenth zero crossing and for every thirteenth zero crossing thereafter to determine the exact time of zero crossover. The time for thirteen of these crossings denotes twelve periods of the wave and this time is presented to a mod-32 counter driven by a simulated digital clock at a 2.424 megacycle rate. The output of this counter is a number between 0 and 31 and occurs every 720 microseconds of the data, producing 224 wave measurements every 0.16128 seconds. The integer mod-32 numbers are written on magnetic tape for input to the third stage of the WPP.

B. Wave, Detailed

The wave period measurement program, WAVE, generates a signal wave with a repeat cycle of 0.16128 seconds. Gaussian random noise is combined with the signal according to specified input signal-to-noise ratios. A twelve period interval of the resulting wave is measured every 720 microseconds, truncated to integers, taken mod-32, and written as output. This produces 224 measurements per cycle for each S/N ratio, of which 12 were used in the first run of the WAVE program. The result is 224 x 12 = 2688 measurements per output record and, considering 101 cycles, or 101 distinct noise blocks, 101 x 2688 = 271,488 total wave measurements for one run of the WAVE program.

The 16 seconds of narrow band gaussian noise from the NOISE program provides the input to the wave period measurement simulation program. The WAVE program adds this noise data to different levels of signals to generate the signal plus noise waves at signal-to-noise ratios of -∞, -6, -4, -3, -2, -1, 0, +1, +2, +3, +4, and +6 db for the first run of WAVE, the first input S/N ratio being noise only.

The signal is a linear FM slide generated as follows: A frequency modulated signal can be expressed

$$a(t) = A \cos \theta (t),$$

where the amplitude A is kept constant and the instantaneous frequency of the cosine function is varied in accordance to the signal. Instantaneous frequency is defined as

$$\frac{\theta(t)}{2\pi} ,$$

which, for a linear FM slide, may be expressed as

$$F + \Delta Ft$$

where

F = carrier frequency = 20,000 hz

ΔF = peak frequency deviation per slide time

Integrating the above expression with respect to time yields

$$\theta(t) = 2\pi \int (F + \Delta Ft)dt = 2\pi(Ft + \Delta Ft^2/2)+\theta_o$$

For the WPP, the peak frequency deviation is computed as

$$\Delta F = \frac{\text{Frequency bandwidth}}{\text{Slide transmission time}} = \frac{440 \text{ cycles/sec.}}{0.16128 \text{ sec.}} = 2728 \text{ hz/sec}$$

Therefore, assuming unity amplitude and zero phase, the linear FM signal can be represented as

$$(1) \quad S = \cos 2\pi(Ft + \Delta Ft^2/2) = \cos 2\pi(20{,}000t + 1364t^2)$$

The signal has a repeat cycle of 0.16128 seconds or 161,280 microseconds. A sample time of 10 microseconds was chosen. This means that the strength of the signal (and the random noise) at the end of every 10 microsecond interval, i.e., at 16128 points, must be calculated. As input to the program, there are 101 blocks of 16128 points of white noise, allowing the generation of 101 different signal plus noise waves from the one 0.16128 second signal wave. This is done for each of the 12 signal to noise ratios specified.

In the generation of the signal with equation (1) above, each signal point J may be computed

$$S(J) = \cos 2\pi(20{,}000t + 1364t^2)$$

where

$$t = 10 \ \mu\text{sec} \ (J - \frac{16129}{2})$$

Initially in the WAVE program, a set of constants are computed for amplitudes of 1 db, 2 db, 3 db, 4 db, 6 db, and others not used in the first WAVE run. A set of 12 scale factors L are computed for the 12 S/N ratios desired, using the db constants, where

$$\text{SCALE}(L) = \text{RMS}x\sqrt{2} \text{ x DB Constant}$$

RMS is the root mean square value of the noise computed during generation of the noise and is injected as a correction factor. The noise and the signal are then combined in the following way:

$$SPN(J) = NOISE(J) + SCALE(L) \times SIGNAL(J)$$

for signal-to-noise ratio L.

The signal may be combined with the noise only at specified intervals to obtain a more realistic waveform. An explanation may be given by example of the first program run at NUWC. Signal was combined with noise for each third noise block so that, for each S/N ratio, two noise blocks (16128 samples per block) were interspersed between each signal plus noise block. Therefore, for each noise or signal plus noise block

$$SPN_I$$

where

I = S/N ratio from 1 through 12,

remembering S/N ratio (1) = $-\infty$ or noise only, the following arrangement is generated:

Block #	1	2	3	4	5	6	7	8 ----	99	100	101
S/N Ratio #											
1	SPN_1	SPN_1	SPN_1	SPN_1	SPN_1	SPN_1	SPN_1	SPN_1----	SPN_1	SPN_1	SPN_1
2	SPN_2	SPN_1	SPN_1	SPN_2	SPN_1	SPN_1	SPN_2	SPN_1----	SPN_1	SPN_2	SPN_1
3	SPN_3	SPN_1	SPN_1	SPN_3	SPN_1	SPN_1	SPN_3	SPN_1----	SPN_1	SPN_3	SPN_1
⋮											
12	SPN_{12}	SPN_1	SPN_1	SPN_{12}	SPN_1	SPN_1	SPN_{12}	SPN_1----	SPN_1	SPN_{12}	SPN_1

The actual wave period measurements are made by examining the signal plus noise samples of each above wave separately for a change in sign from minus to plus. The simulation detects 13 successive positive zero crossings and, if the first and thirteenth crossing points are not identically zero, an interpolation is made on those two points, assuming a sine wave, to obtain more accurate crossing times. The time elapsed between the first and thirteenth crossings determines 12 periods of the signal plus noise wave. 720 microseconds (or 72 - 10 microsecond samples) are added to the first crossing time and the above process is repeated. Thus, 224 elapsed time measurements (called TT's in the program) will be obtained every 0.16128 seconds.

Each of the elapsed time measurements are then subjected to a modular 32 counter driven by a simulated 2.424 megacycle clock. Also injected is a sliding preset which is set to zero at the start of each 224 FM slide transmission and is preset to the next higher counter state every seventh sample period (i.e., every 7 TT's). In an actual system, however, the counter may start in any arbitrary position. This effect is remedied by an averaging process in the third program. The elapsed time measurements (TT's) are therefore combined with the clock and the sliding preset in the manner

TT x CLOCK + PRESET,

truncated to integers, and taken modulo-32. The resulting integer numbers between 0 and 31 (denoted IT's by the program and having a one to one correspondence with the TT's) are written as output on magnetic tape.

Symbol Correspondence

COS Library routine for trigonometric cosine of a real argument

DELT Scale factor, 10^{-5}, used in various calculations; also, 10 microsecond constant

DBIPHF Constant value for 1 1/2 db (not used in first run)

FO Carrier frequency (20,000 kHz)*

F1 One-half of the peak frequency deviation: 1/2(bandwidth/cycle time) = 1/2(440.0/0.16128)

F3DB Constant value for 3 db

F4DB Constant value for 4 db

F5DB Constant value for 5 db (not used in first run)

F6DB Constant value for 6 db

F12DB Constant value for 12 db (not used in first run)

F18DB Constant value for 18 db (not used in first run)

F24DB Constant value for 24 db (not used in first run)

F30DB Constant value for 30 db (not used in first run)

FBW Signal bandwidth (440 Hz)

FNP Floating point value ** for the number of positive zero crossings for one wave measurement (number of wave periods for each measurement)

FNUMBR Floating point value of number of noise samples in one block; also the slide transmission time X 10^{5} or the repeat cycle X 10^{-5}

* Numbers in parenthesis refer to values used in first NUWC run.

** Mixed mode arithmetic is not available; therefore, constants must be entered in any and all modes in which they are used.

ICROSS Index for <u>NP</u> positive zero crossings (12)

INT Library routine for truncation of a real argument to an integer

IT Modulo - 32 integer wave period measurements computed from TT's

ISET Sliding preset combined with TT's in forming modulo - 32 IT's

J General index; also, current position within noise block when computing positive zero crossings

L2,L6 Previously * were status locations for NTRAN operations which wrote IT's on magnetic tape; Equal to LTT for successful operation. No longer used.

LNGTH Intermediate number of noise points generated equalling $\frac{1}{MULTP}$ of one block of noise (288). Originally * equal to number of noise samples on one magnetic tape record.

LOUT Number of wave period measurements per cycle (224).

LSCALE Number of signal/noise ratios to consider; therefore, number of scale factors to calculate and the number of S/N ratios to calculate for each noise block (12).

LTT Number of wave measurements per cycle X number of S/N ratios (224 X 12 = 2688); Size of one magnetic tape record of IT's

* Refers to Sperry program format.

MLTM1 Equal to MULTP-1 for indexing purposes (55)

MOD Library routine for remaindering integer arguments

MULTP Multiplication factor which, along with LNGTH, determines the length of one block of noise (56)

NBLKS Maximum number of noise blocks to be used for one WAVE run

NP Number of positive zero crossings for one wave period measurement (12)

NUMBR Number of samples in one noise block (16128)

ONEDB Constant value for 1 db

OSCL Digital clock frequency calculation $(2.424 \times 10^6$ cycles) = (2.424 megacycles)

PI $\pi(3.1415927)$

PLIN Intermediate calculation for interpolation of positive zero crossing; Equal to 0^+value/(0^+value - 0^-value)

RMS Root mean square of noise used as a correction factor for S/N ratio scale factors; Equal to the standard deviation for the total noise run (see SIGZ in program KAVEE)

S Signal value

SCALE Array containing scale factors for Signal + Noise computations

SJ Floating point value for J when used as the position index within the noise block

SPN Temporary storage for one block of noise; also signal is combined with the noise for signal + noise values here

SQRT2 Numerical value for the square root of 2.0 (1.4142136)

START First positive zero crossing and therefore the start of one set of 224 wave period measurements

TENDB Constant value for 10 db (not used in first run)

THRDB Constant value for 3 db

TNUMBR Intermediate calculation of signal relating to the number of noise samples in one block (16129/2)

TP Repeat cycle (0.16128 seconds)

TSCALE Digital clock frequency scaled 10^{-5}

TT A twelve period wave measurement of elapsed time in microseconds

TWODB Constant value for 2 db

TX Intermediate calculation of signal

WAVE Name of wave period measurement program

Input

One IBM compatible magnetic tape with 101 blocks of low bandpass random noise, 16128 samples in each block.

Output

Each noise block input is processed with each of the 12 S/N ratios, and one output wave record consists of 224 x 12 = 2688 mod-32 integer wave period measurements. There will be 101 such 2688 word records written on an IBM compatible magnetic tape.

After each record is written, the index (1-101) is printed out.

Subroutines

Library functions called are:

COS	Computes trigonometric cosine of a real argument
INT	Truncates a real argument to an integer
MOD	Computes remainder of two integer arguments

Execution

The entire program deck is loaded, since Fortran IV at NUWC-SD is an automatic compile and execute system. A listing of the WAVE program is enclosed. A PAUSE has been inserted to stop the program at the beginning of execution so that the noise input tape and the output tape may be mounted.

Tape Assignments for NUWC-SD Univac 1230 or AN/USQ-20 computers

System tape on Unit M1

Noise input tape on Unit M2

Scratch tape on Unit M3

Wave output tape on Unit M4

Execution Time

To generate one record of wave data of 2688 words takes 2 1/2 hours on the Univac AN/USQ-20 computer and 5 minutes on the Univac 1230 computer. However, in order to take advantage of the fast double precision hardware on the 1230 computer, the entire program was converted to double precision, with an average running time of 3 minutes per record.

Accuracy

The wave program was written in single precision since the final output is integer and requires no greater precision. However, as stated above, the entire program was also written in double precision to take advantage of the fast floating point hardware on the Univac 1230 computer. The program in both forms is included in Appendix C.

VI. PROCS SUBROUTINE

A. PROCS, General

The PROCS Routine commences, as related to hardware, with the zone logic circuits. See Figure 2. The function of the zone logic circuits is to optimize the processing. In actual practice the zone logic circuits function as band pass filters. Each zone is 200 Hz wide and they are displaced by 100 Hz.

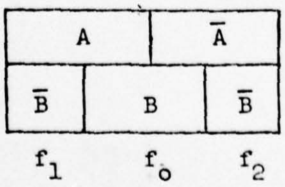


Figure 2. ZONE LOGIC SPECTRUM

The input to the zone gate logic block are numbers 0 to 31. These numbers represent a frequency and in the program are denoted as IT's. The zone logic circuits are simulated by dividing the numbers 0-31 into zone A, B, $\bar{A}$ and $\bar{B}$. An "A" channel is generated for an IT between 0-15 and a B channel for an IT between 8-23 inclusive. A moving sum of 224 samples is taken for each channel and its complement. i.e.,

$$\text{sum}_1 = \Sigma \text{ ZONCT}$$

$$\text{sum}_i = \text{sum}_{i-1} + \text{ZNCOUNT}_{i+223} - \text{ZONCT1}_i$$

This moving sum is denoted as ANUM and its length (224 samples) represents the signal period.

As stated on the description of the hardware system, the digitizer is preset in such a manner that a coherent signal will appear as stationary frequency. Therefore, ANUM represents the number of times a particular frequency falls on a zone and is a measure of period correlation during a signal period. The ANUMS for channels A and B are compared and the maximum selected to denote the ANUM of that particular sample. The ANUM's are then range peaked detected by selecting the maximum ANUM every 672 (range bin) ANUMS in a specified range bin. In the hardware system these ANUMS are the output of the WPP. In the simulation program these numbers are used to plot probability of exceeding threshold vs threshold with input S/N as a parameter.

B. PROCS, Detailed

The PROCS subroutine is divided into four phases.

Phase 1

The PROCS routine begins by reading the number of NRUNS, LSCALE, and IPEAK; then it positions the tape to the file specified by the LSCALE. The input to this routine are numbers called IT's and are stored by files on magnetic tape. Each file represents a particular input signal to noise ratio. Due to the core limitation only 11,424 of the IT's are read at first. The subroutine ZONEIT is called and each IT is ZONED.

The subroutine ZONEIT defines 8 separate zoning conditions where the zones are shifted to cover one more number to the right and one less number to the left, and where zones A, $\bar{A}$, B, $\bar{B}$ each time correspond to a different set of numbers between 0-31. A sum of the number of times it falls into a zone is kept. These sums, one for Zone A and one for Zone B are divided by 8 and the results denoted as AIT or BAIT.

The output for the ZONEIT subroutine is denoted AIT and BAIT for each IT, and these are recorded in records of 224 on magnetic tape units M1 and M2 respectively.

Phase 2

After the remaining IT's are processed, M1 is rewound and again only half of the AITS are now read and stored. In this phase the moving sample sum for every 224 samples is computed and if less than 112 then the sum is complemented. If complementation takes place, in effect, the $\bar{A}$ is the largest value and retained to represent that zone. The moving sample sum is denoted as ANUMA and can be expressed as:

$$\text{ANUMA}(1) = \sum_{i=1}^{224} \text{AIT}_i$$

YC(2)	30 - 52	D23.8
YC(3)	56 - 78	D23.8

*The second-stage values would be entered through E w.d field for the primarily single precision noise program.

$$ANUMA(i) = -AIT(i) + ANUMA(i-1) + AIT(i+223)$$

Every 224 samples the ANUMA's are written on M2 following the BAITS. The remaining AIT's are processed and recorded. The BAITS are processed identically except the ANUMS are recorded on M1.

Phase 3

In the third phase the maximum value between ANUMA and ANUMB is chosen to represent that sample. The data is read from M1 and M2 and the output denoted as ANUM is written on M3.

Phase 4

The fourth phase performs three functions:

1. Performs peak detection
2. Computes the statistics
3. Prints results

Range peak detection is accomplished by reading in from M3 through ANUM and choosing the largest value every 672 samples. At present each file of data contains 33 signal periods; therefore there will be 33 values chosen. The computation of statistics is accomplished by obtaining a distribution of the ANUM's; i.e., it will count the number of ANUMS that fall into 113 theoretical bins numbered 112-224 and denoted as ISTAT. Each bin represents a threshold value. Also it computes the percentage of total that an ANUM falls within each bin and all lower bins, i.e.,

$$\text{Percentage } (\%) = \sum_{i=1}^{32} \text{ISTAT } (i)$$

The above computation, in essence, is cumulative distribution of the ANUM's, and the results shown in Figure 3 through 14 are a pictorial representation of the printed results. Flow charts are shown in Appendix B. The program in its present form utilizes approximately 54000 OCTAL cells including library subroutines.

Inputs/Outputs

1. IT's (mag tape) 22400 or 22624
2. LSCALE (punched card) specifies input S/N ratio
3. IPEAK (punched card) specifies range peak detection
   a. 1 no peak detection - print distribution only
   b. 2 no peak detection - print distribution
   c. 3 peak detection in range (every 672 samples) and print distribution

VII. CONCLUSIONS

1. The primary objective to adapt the WPP simulation program to NUWC facilities was accomplished.

2. In order to obtain a clearer comparison between NUWC and Sperry's results it was decided to triple the input data to obtain greater resolution and smoother curves. The results obtained with the adapted program are nearly identical to the results Sperry obtained. Figures 3 through 14 show the results for input signal to noise ratios (DB) of -∞, -6, -4, -3, -2, -1, 0, +1, +2, +3, +4, +6, respectively, for both the adapted and the original program.

3. Presently the program is being used to investigate processing gain, and modification to the program is underway to investigate range binning effects on processing gain. Plans are being made to modify the program to investigate (1) Mutual Ship Interference, (2) Structured Echo Analysis.

VIII. ACKNOWLEDGEMENTS

The Data Processing Program Generation and Process Simulation Group, Code D5503, NUWC-SD, was especially helpful in its timely implementation of Fortran IV on the Univac 1230 and AN/USQ-20 computers, making the conversion of the Sperry Gyroscope Company Univac 1107 programs a much simpler task.

References

"WPP Simulation Study for PAIR Sonar Systems", October 13, 1967, Sperry Gyroscope Division, Sperry Rand Corporation, Great Neck, New York.

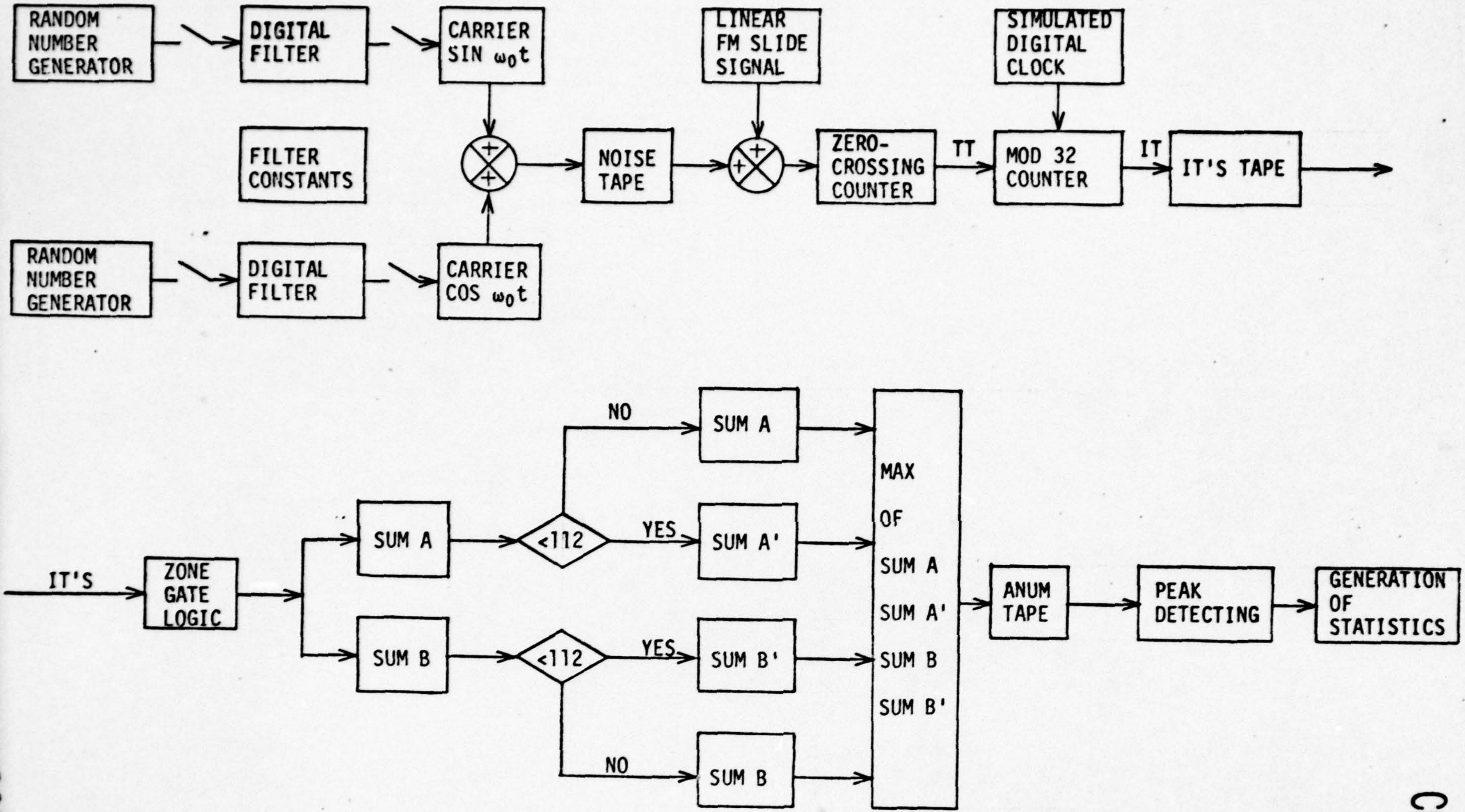


FIGURE 1: BLOCK DIAGRAM WAVE PERIOD PROCESSOR DIGITAL SIMULATION

CONFIDENTIAL

CONFIDENTIAL

INPUT SIGNAL/NOISE -∞ DB

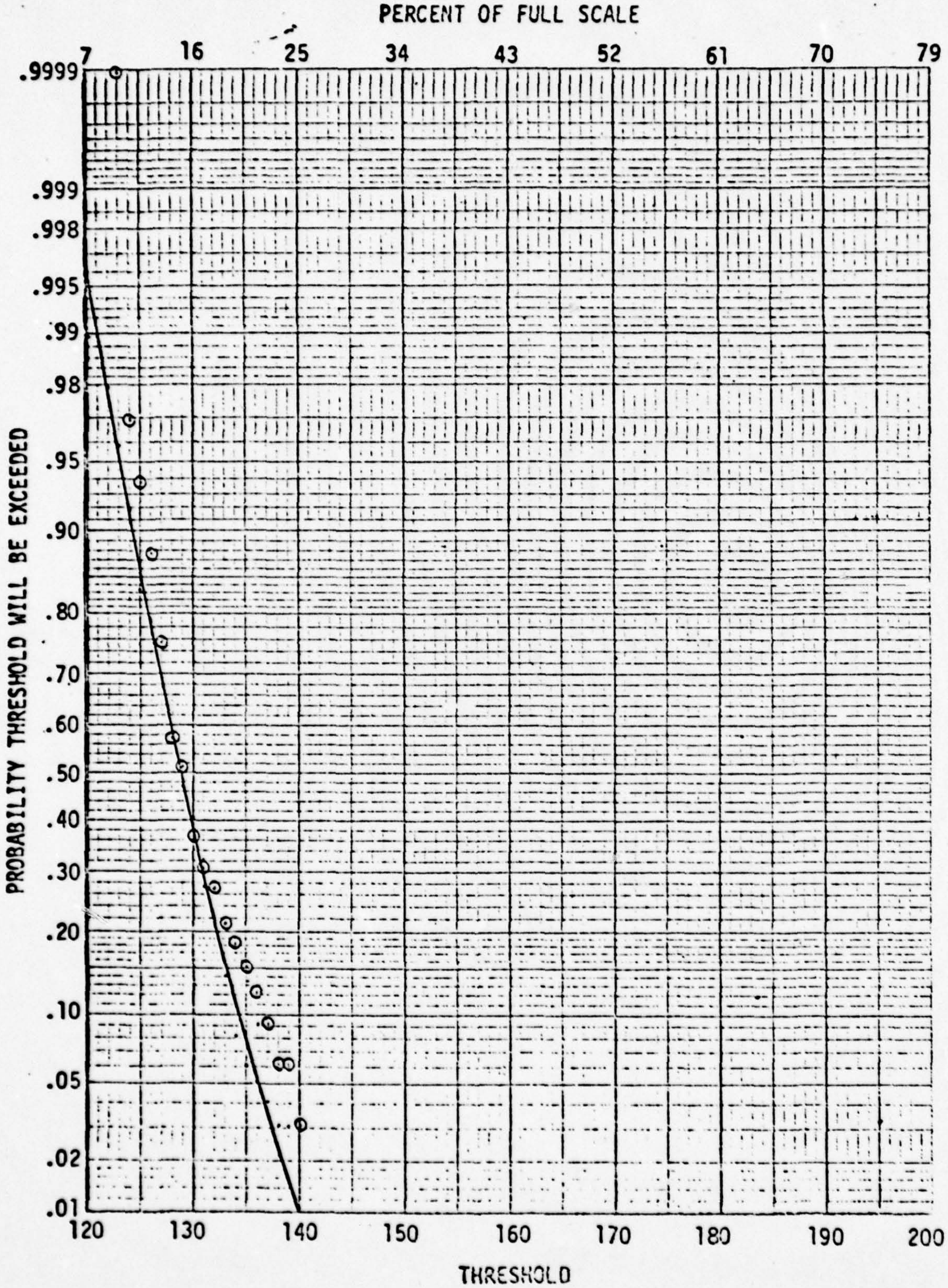


FIGURE 3: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

——	NUWC
⊙	Sperry Rand

INPUT SIGNAL/NOISE -6 DB

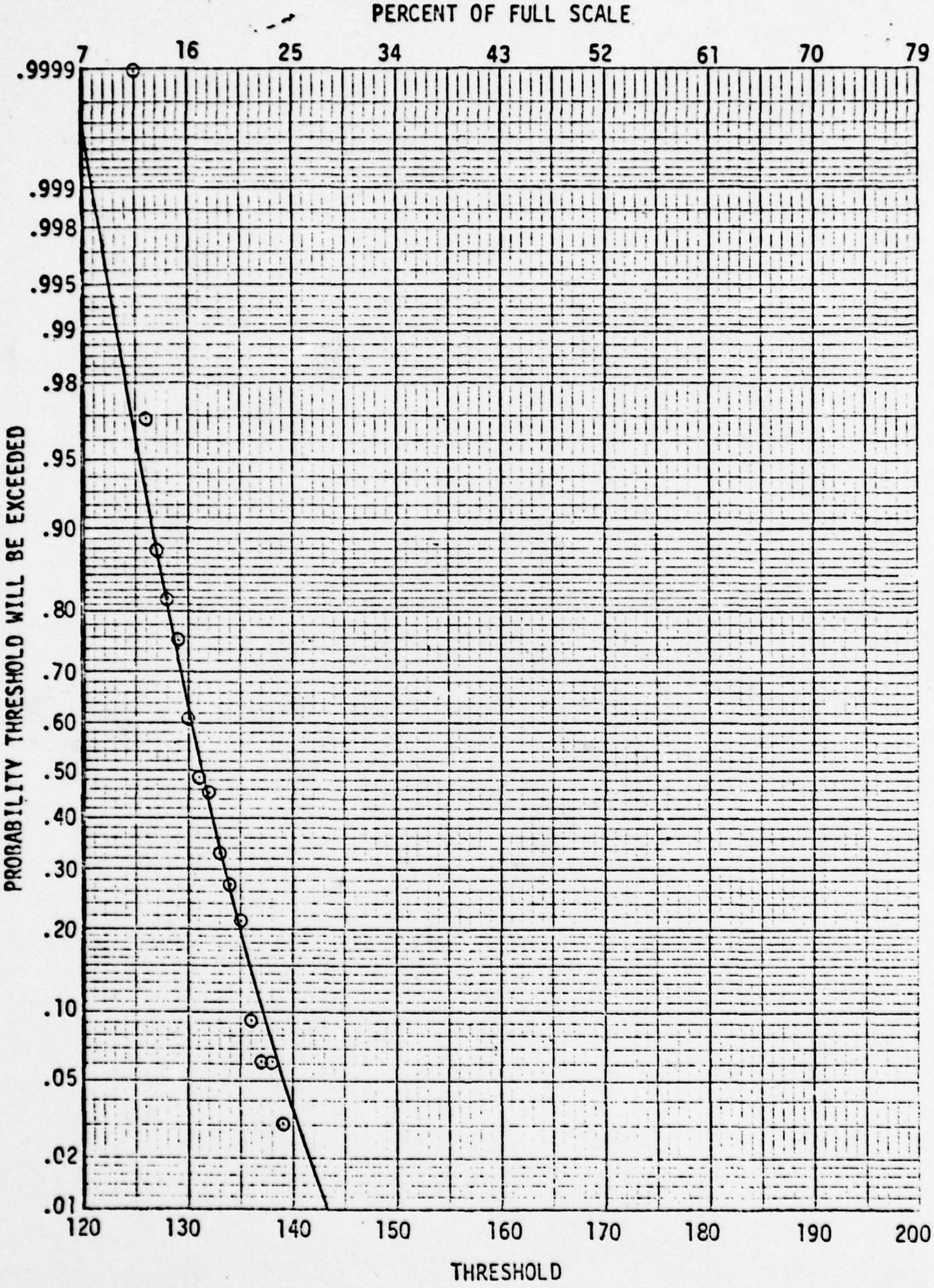


FIGURE 4: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

CONFIDENTIAL

INPUT SIGNAL/NOISE -4 DB

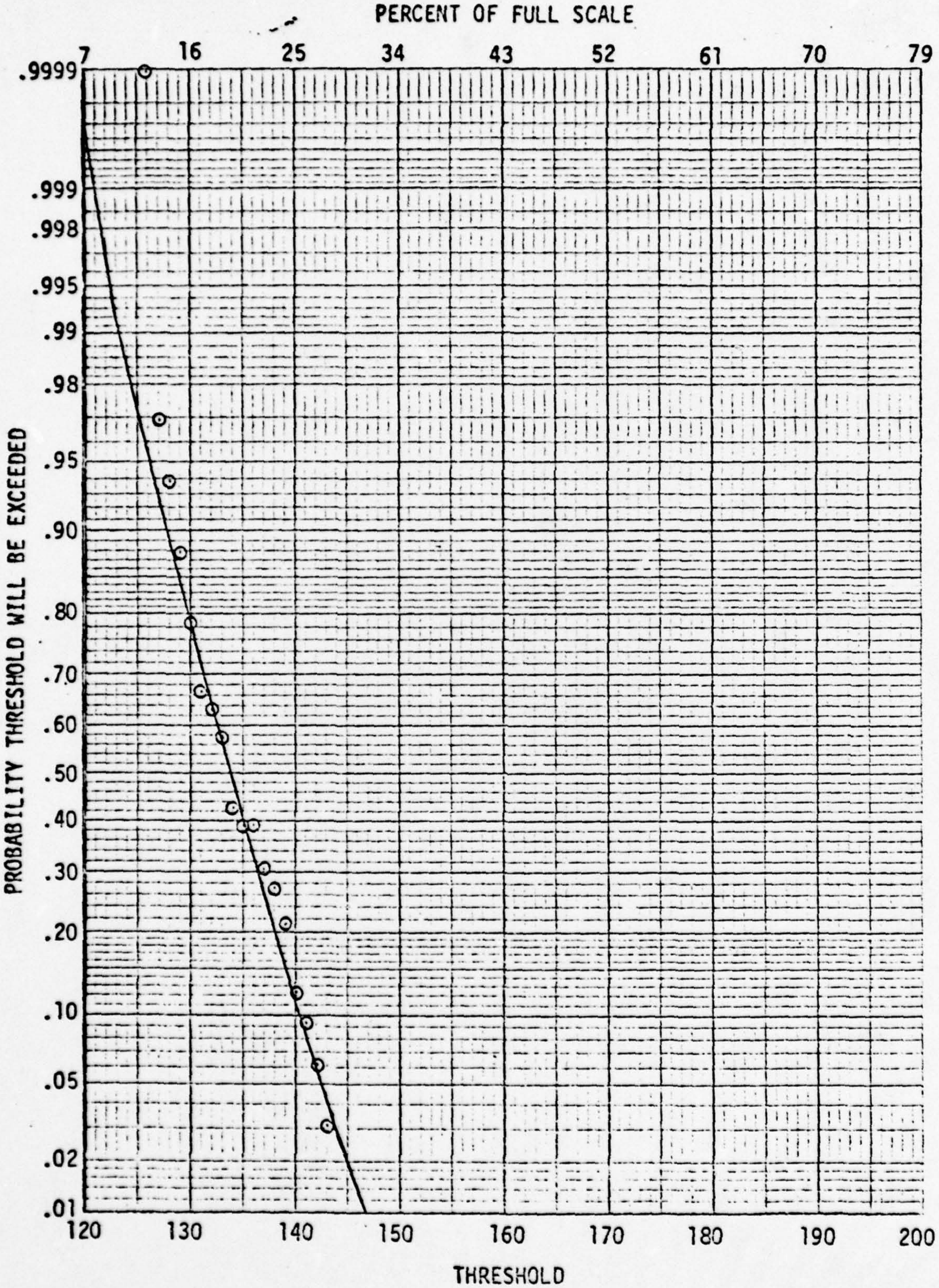


FIGURE 5: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

CONFIDENTIAL

INPUT SIGNAL/NOISE -3 DB

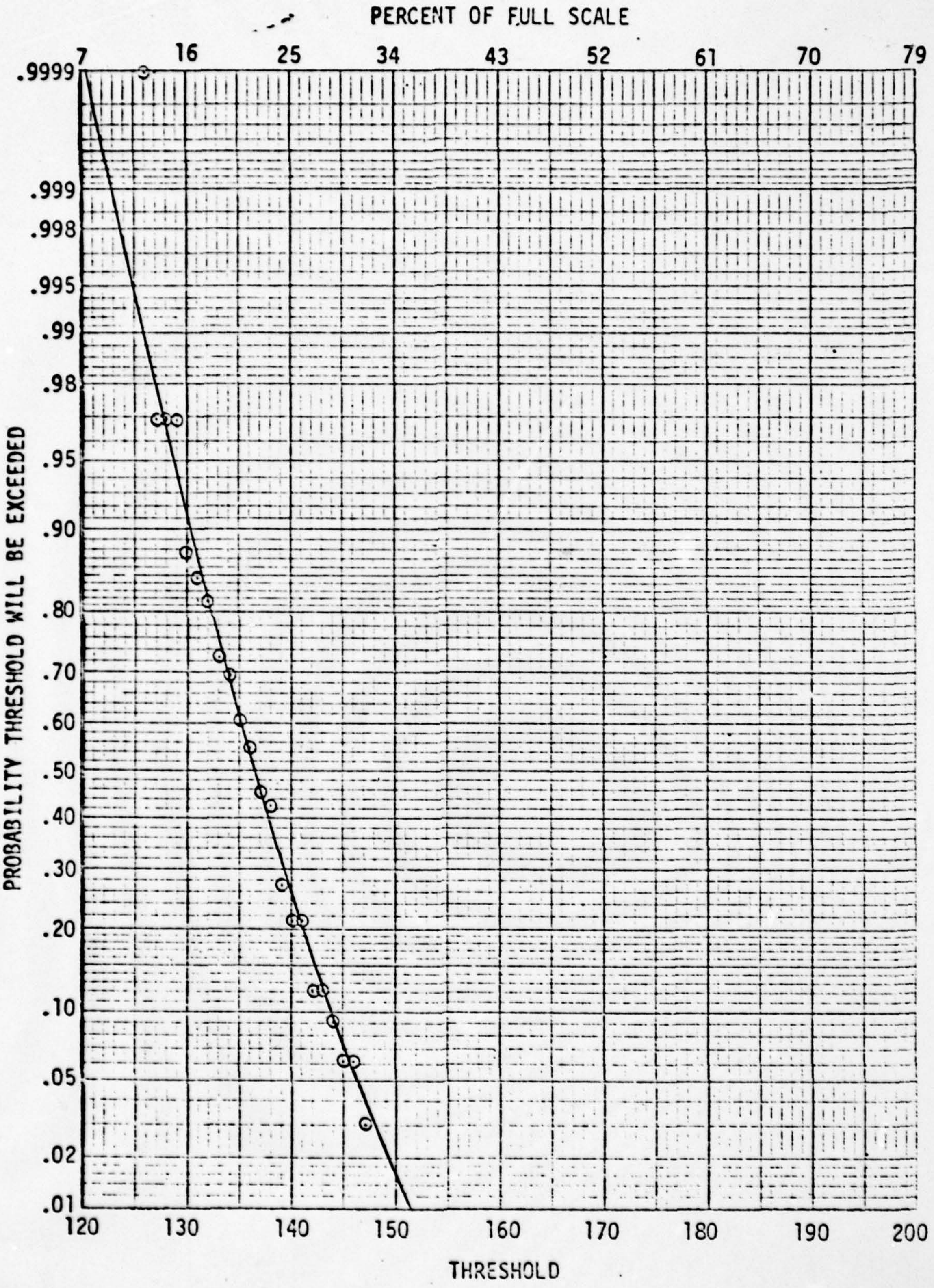


FIGURE 6: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

——— NUWC
ʘ Sperry Rand

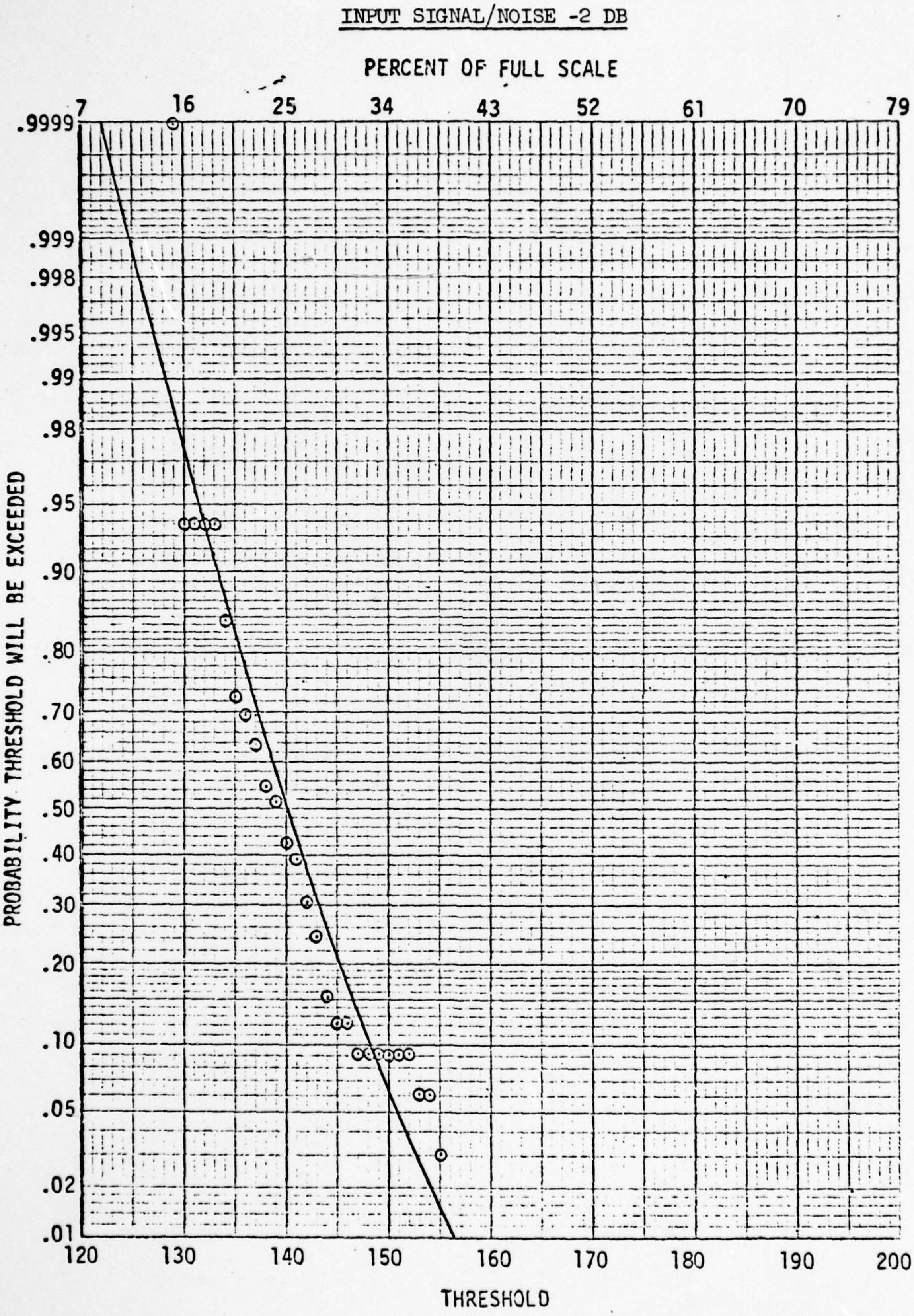


FIGURE 7: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

INPUT SIGNAL/NOISE -1 DB

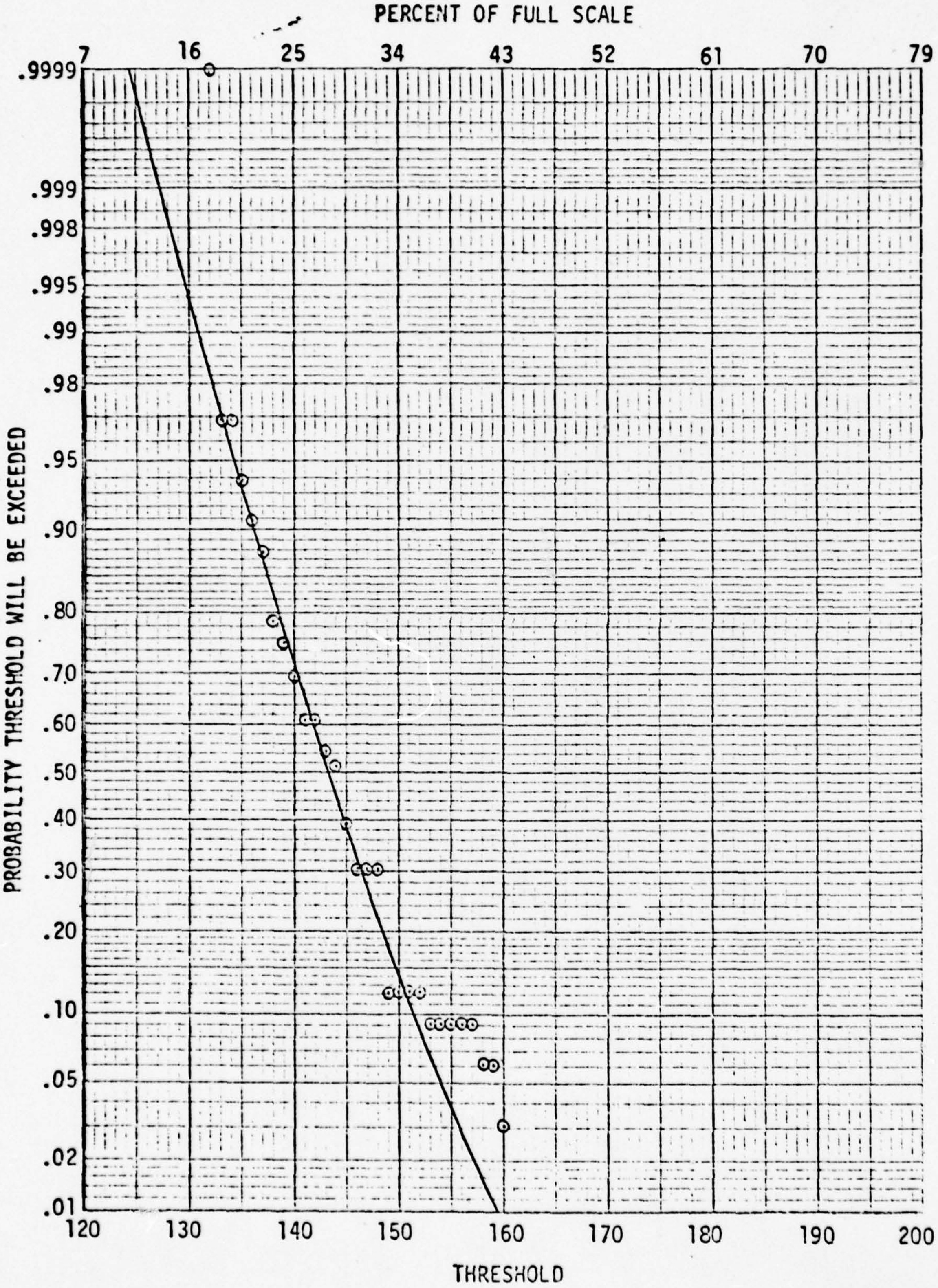


FIGURE 8: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

INPUT SIGNAL/NOISE 0 DB

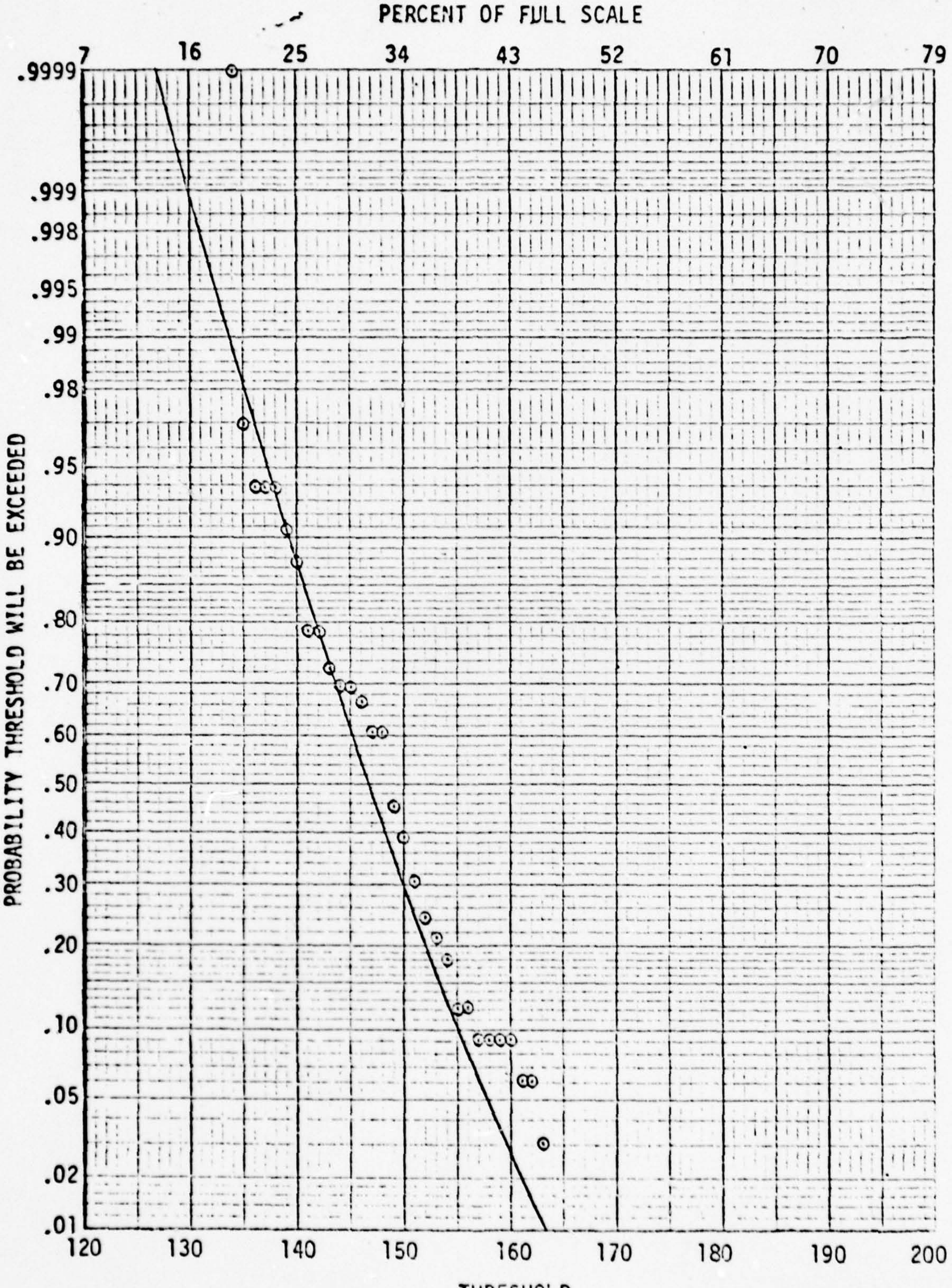


FIGURE 9: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

——— NUWC
ʘ Sperry Rand

CONFIDENTIAL

INPUT SIGNAL/NOISE +1 DB

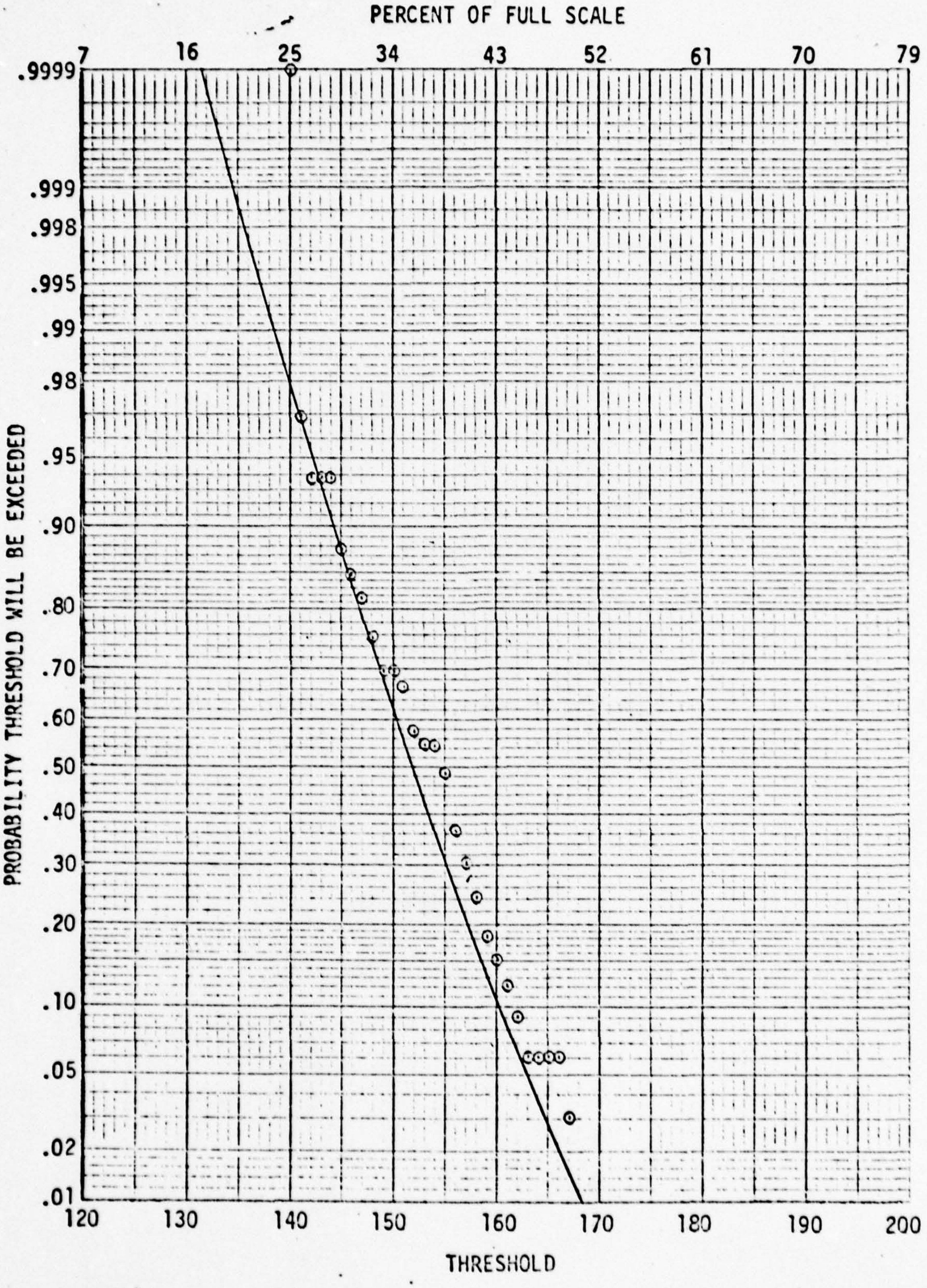


FIGURE 10: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

CONFIDENTIAL

INPUT SIGNAL/NOISE +2 DB

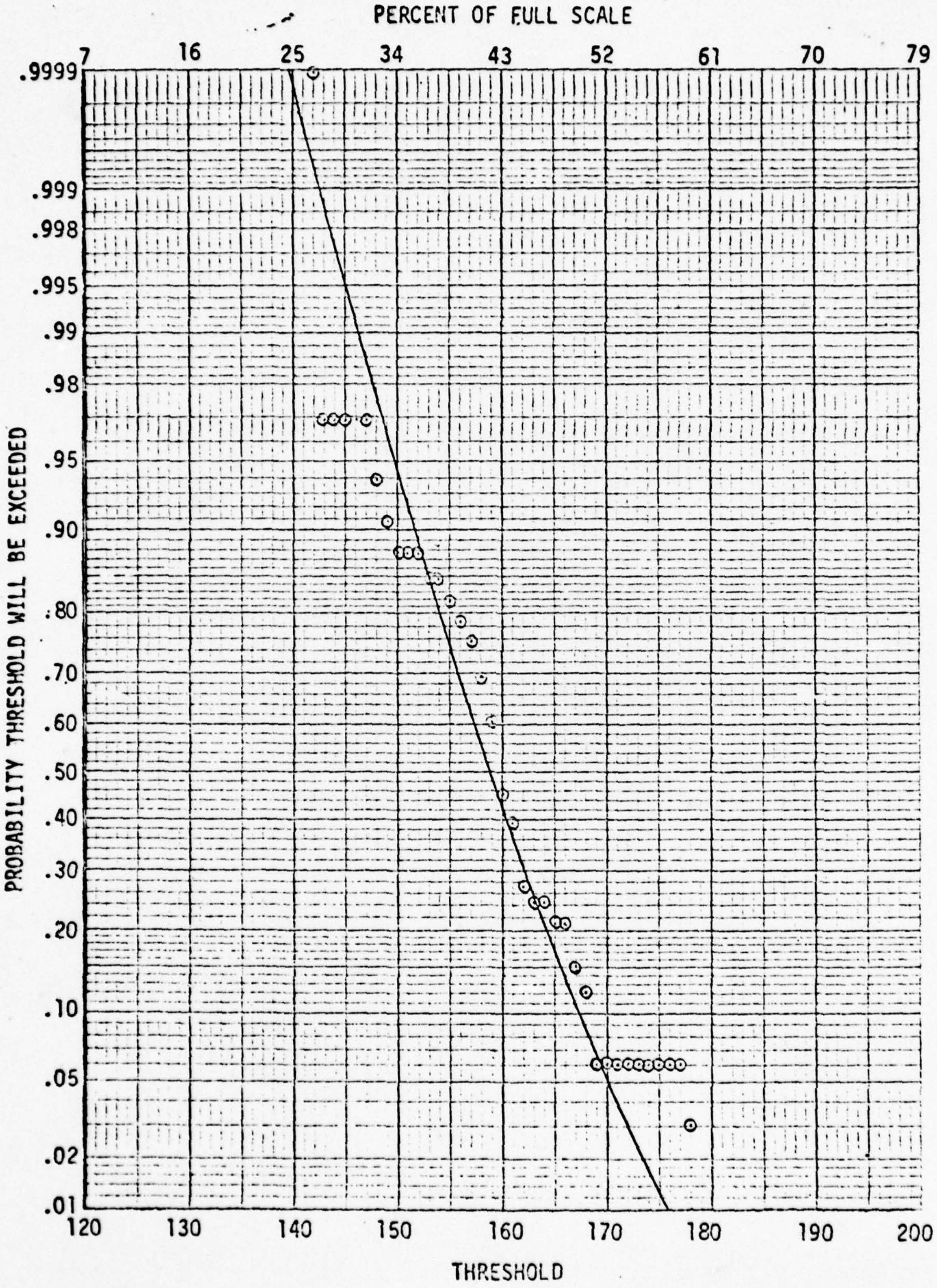


FIGURE 11: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

CONFIDENTIAL

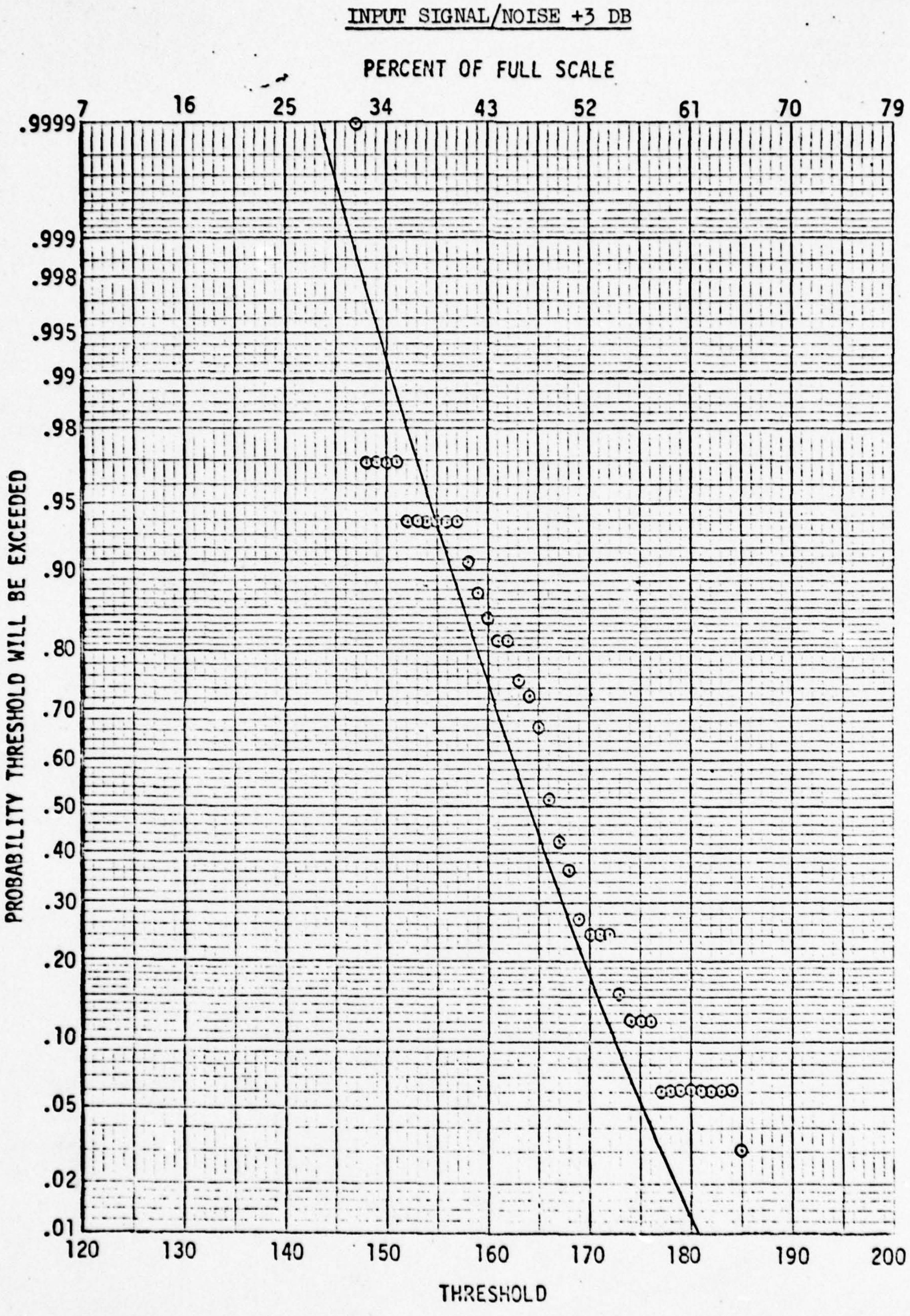


FIGURE 12: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
⊙ Sperry Rand

CONFIDENTIAL

INPUT SIGNAL/NOISE +4 DB

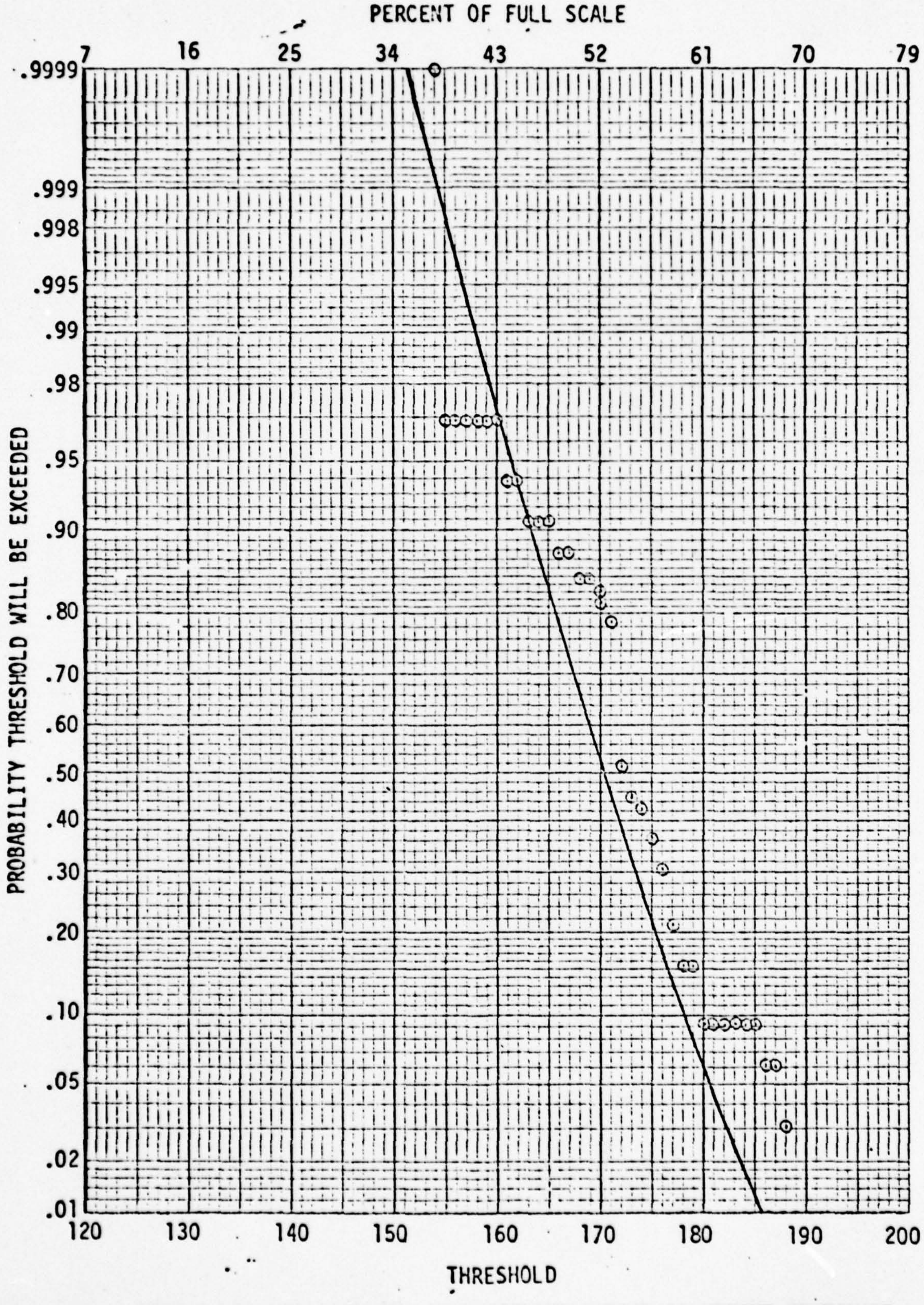


FIGURE 13: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC

⊙ Sperry Rand

CONFIDENTIAL

INPUT SIGNAL/NOISE +6 DB

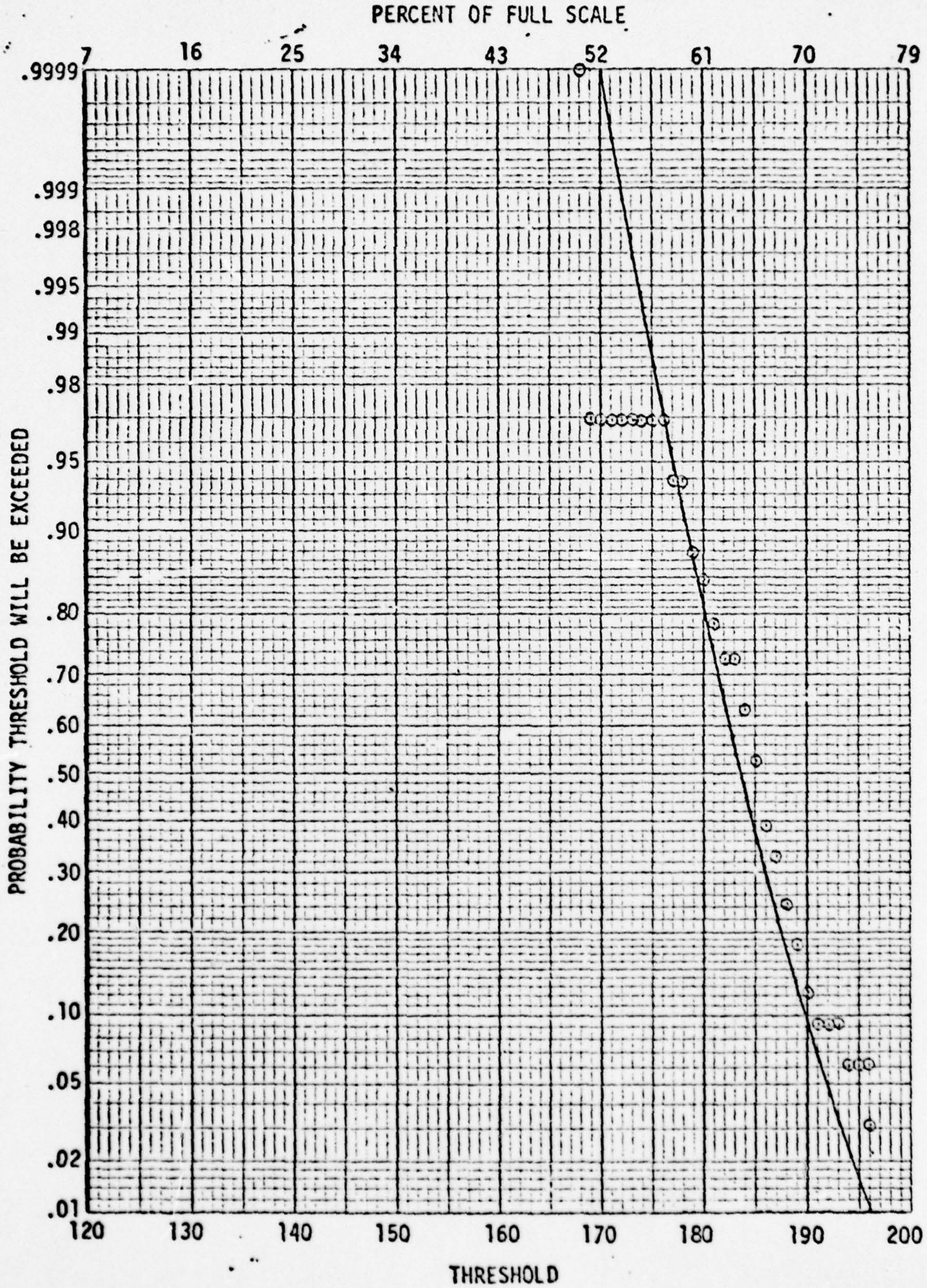


FIGURE 14: PROBABILITY OF EXCEEDING THRESHOLD VERSUS THRESHOLD

—— NUWC
ʘ Sperry Rand

CONFIDENTIAL

APPENDIX A

RANDM, GAUSS

RANDM1 - Uniform Pseudorandom Number Generator

This function subprogram generates random numbers whose density function is uniform on the interval 0.0 to 1.0 and zero elsewhere. The program is written in CS-1 phase I mono-code language with special linking sequences for Fortran IV. There are three entry points to the generator (Fortran IV calling sequences):

R = RANDM1 (Dummy)

generates one number and stores it in R

Call RANDM2 (L)

puts the integer L into the starting point of the generator

L = IRAND3 (Dummy)

puts the current integer starting point of the generator into L

This random number generator was adapted from the method suggested by D. H. Lehmer, explained in the article "A New Pseudorandom Number Generator" by David W. Hutchinson in the Communications of the ACM, Vol. 9, Number 6, June 1966. The Lehmer generator is:

$$X_{i+1} = AX_i \pmod P$$

where P is the largest prime less than 2^{29} and A is a primitive root of P.

The generator RANDM1 uses $P = 2^{29} - 3 = 536,870,909$ D and $A = 5^5 = 3125$ D. The numbers are generated by:

1. Multiply the starting value by a constant (5^5).
2. Divide the product by the largest prime less than 2^{29}.

3. Store the result as the next starting point.

4. Convert the result to floating point.

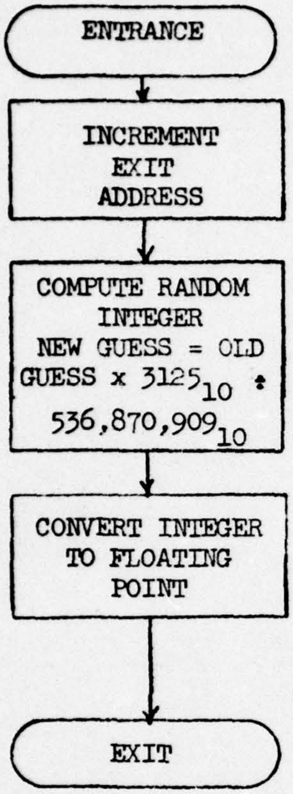
The initial starting value is a fixed point 1.

Notes: 1. The dummy argument in RANDM1 and IRAND3 is necessary but its value and type are unimportant.

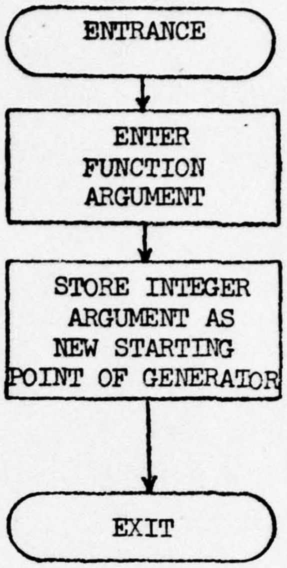
2. A CS-1 output 324 with the relocatable object program on cards is inserted in the Fortran deck just before the $DATA card.

3. The routine occupies 55_8 locations.

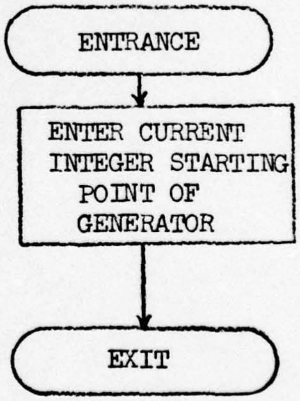
RANDM1


ENTRANCE
INCREMENT EXIT ADDRESS
COMPUTE RANDOM INTEGER NEW GUESS = OLD GUESS x 3125_{10} ÷ $536,870,909_{10}$
CONVERT INTEGER TO FLOATING POINT
EXIT

RANDM2



IRAND3



GAUSS - Pseudorandom Number Generator, Normal Density

This function subprogram generates random numbers whose density function is gaussian with zero mean and unit variance. The method used is a Mueller Box:

Given two independent random numbers, x_1 and x_2, uniformly distributed on the interval 0-1, two independent gaussian-distributed random numbers are obtained via the following algorithm:

$$y_1 = \sqrt{-2 \ln x_1} \quad \cos 2 \pi x_2$$

$$y_2 = \sqrt{-2 \ln x_1} \quad \sin 2 \pi x_2$$

This algorithm was obtained from Memorandum 9, Random Noise Generation Routines, February 1967, prepared by E. C. Fraser, Stanford Research Institute, Project ESU 5830.

The method is the same as that used in the 1107 library routine, given by:

$$f_x (X) = \frac{1}{\sqrt{2 \pi}} \quad \exp (-X^2/2)$$

The new GAUSS routine is written in Fortran IV with the calling sequence:

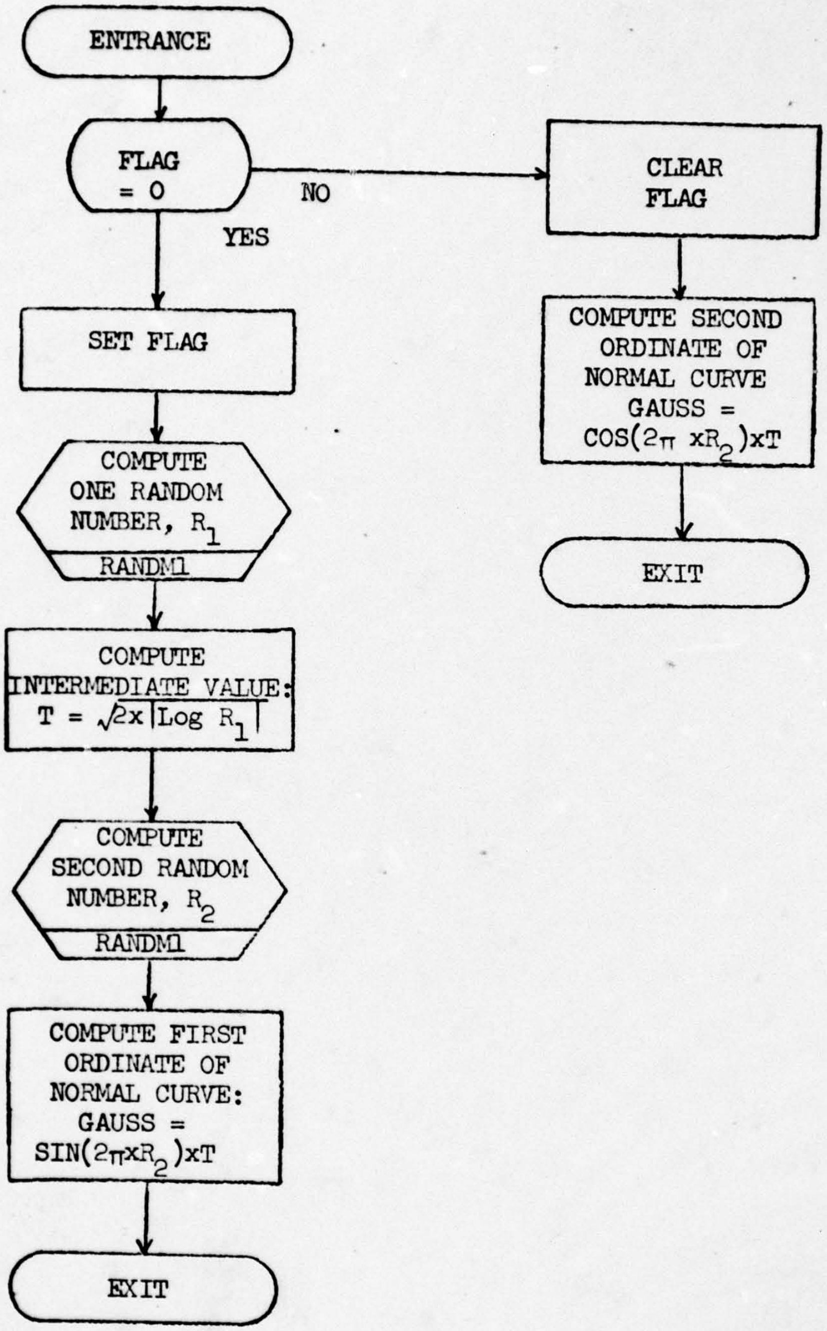
R = GAUSS (Dummy)

Notes: 1. The routine uses the uniform random number generator RANDM1 twice for each two gaussian numbers requested.

2. The routine calls the SQRT, ABS, ALOG, SIN, and COS routines.

3. The routine occupies 102_8 locations.

GAUSS


ENTRANCE
FLAG = 0
NO
YES
SET FLAG
COMPUTE ONE RANDOM NUMBER, R_1
RANDM1
COMPUTE INTERMEDIATE VALUE: $T = \sqrt{2x|\text{Log } R_1|}$
COMPUTE SECOND RANDOM NUMBER, R_2
RANDM1
COMPUTE FIRST ORDINATE OF NORMAL CURVE: GAUSS = $\text{SIN}(2\pi x R_2) x T$
EXIT
CLEAR FLAG
COMPUTE SECOND ORDINATE OF NORMAL CURVE GAUSS = $\text{COS}(2\pi\ x R_2) x T$
EXIT

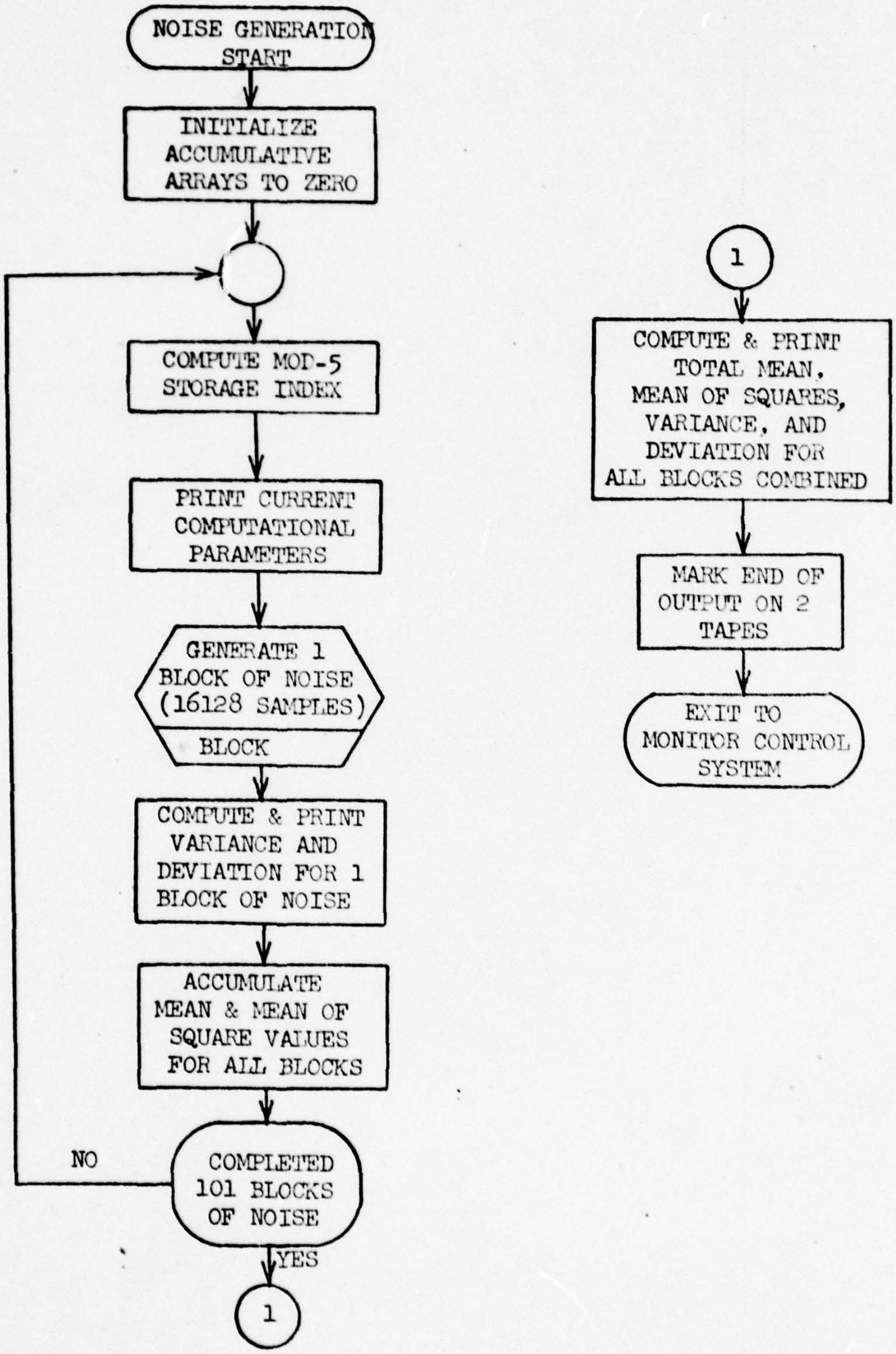
APPENDIX B

Program Flowcharts

FUNCTIONAL FLOWCHART:

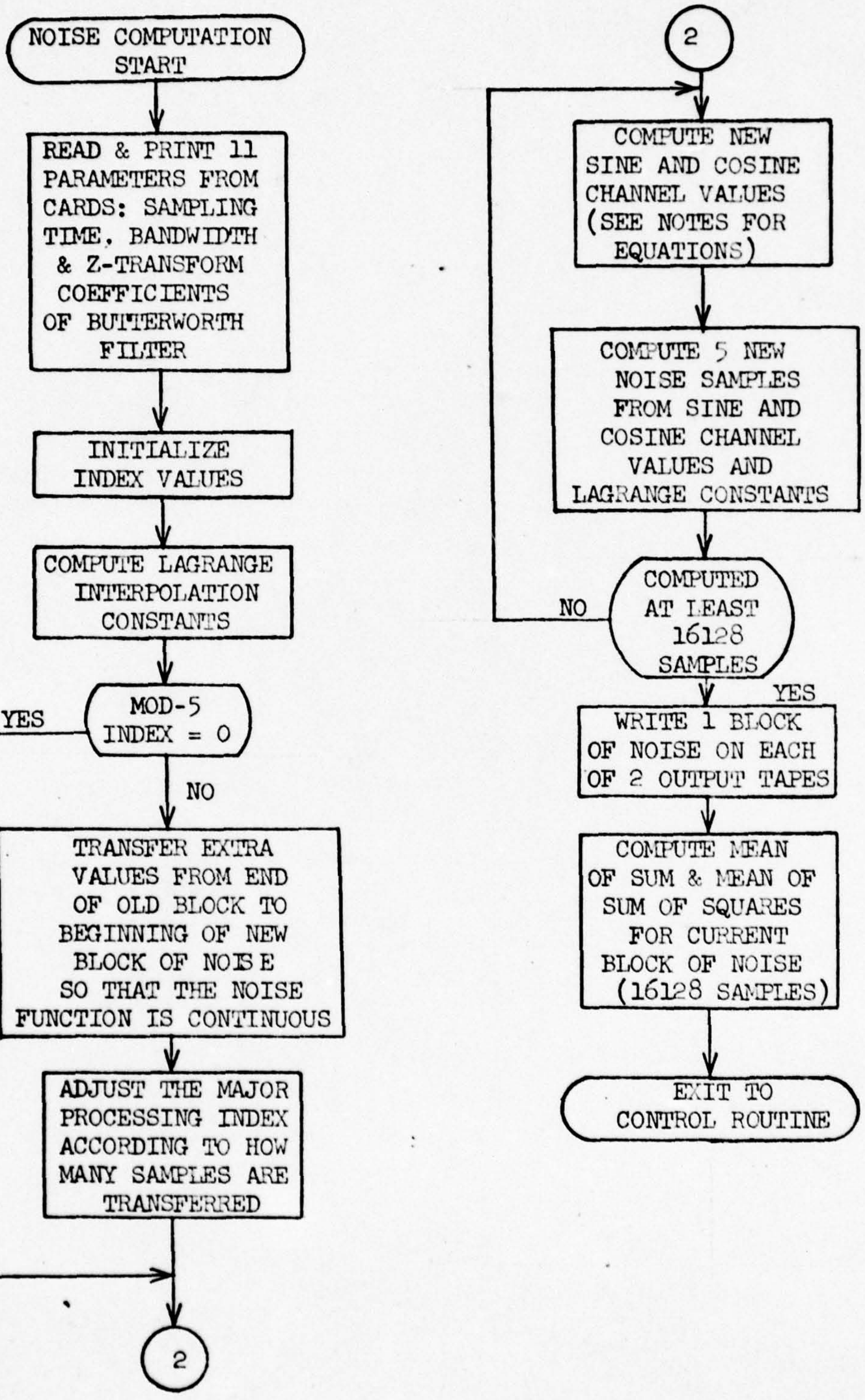
Noise Subroutine

WAVE PERIOD PROCESSOR NOISE GENERATION
FUNCTIONAL FLOWCHART
CONTROL ROUTINE: KAVEE


NOISE GENERATION START
INITIALIZE ACCUMULATIVE ARRAYS TO ZERO
COMPUTE MOD-5 STORAGE INDEX
PRINT CURRENT COMPUTATIONAL PARAMETERS
GENERATE 1 BLOCK OF NOISE (16128 SAMPLES)
BLOCK
COMPUTE & PRINT VARIANCE AND DEVIATION FOR 1 BLOCK OF NOISE
ACCUMULATE MEAN & MEAN OF SQUARE VALUES FOR ALL BLOCKS
COMPLETED 101 BLOCKS OF NOISE
NO
YES
1
1
COMPUTE & PRINT TOTAL MEAN, MEAN OF SQUARES, VARIANCE, AND DEVIATION FOR ALL BLOCKS COMBINED
MARK END OF OUTPUT ON 2 TAPES
EXIT TO MONITOR CONTROL SYSTEM

FUNCTIONAL FLOWCHART

COMPUTATIONAL ROUTINE: BLOCK


NOISE COMPUTATION START
READ & PRINT 11 PARAMETERS FROM CARDS: SAMPLING TIME, BANDWIDTH & Z-TRANSFORM COEFFICIENTS OF BUTTERWORTH FILTER
INITIALIZE INDEX VALUES
COMPUTE LAGRANGE INTERPOLATION CONSTANTS
MOD-5 INDEX = 0
YES
NO
TRANSFER EXTRA VALUES FROM END OF OLD BLOCK TO BEGINNING OF NEW BLOCK OF NOISE SO THAT THE NOISE FUNCTION IS CONTINUOUS
ADJUST THE MAJOR PROCESSING INDEX ACCORDING TO HOW MANY SAMPLES ARE TRANSFERRED
2
2
COMPUTE NEW SINE AND COSINE CHANNEL VALUES (SEE NOTES FOR EQUATIONS)
COMPUTE 5 NEW NOISE SAMPLES FROM SINE AND COSINE CHANNEL VALUES AND LAGRANGE CONSTANTS
COMPUTED AT LEAST 16128 SAMPLES
NO
YES
WRITE 1 BLOCK OF NOISE ON EACH OF 2 OUTPUT TAPES
COMPUTE MEAN OF SUM & MEAN OF SUM OF SQUARES FOR CURRENT BLOCK OF NOISE (16128 SAMPLES)
EXIT TO CONTROL ROUTINE

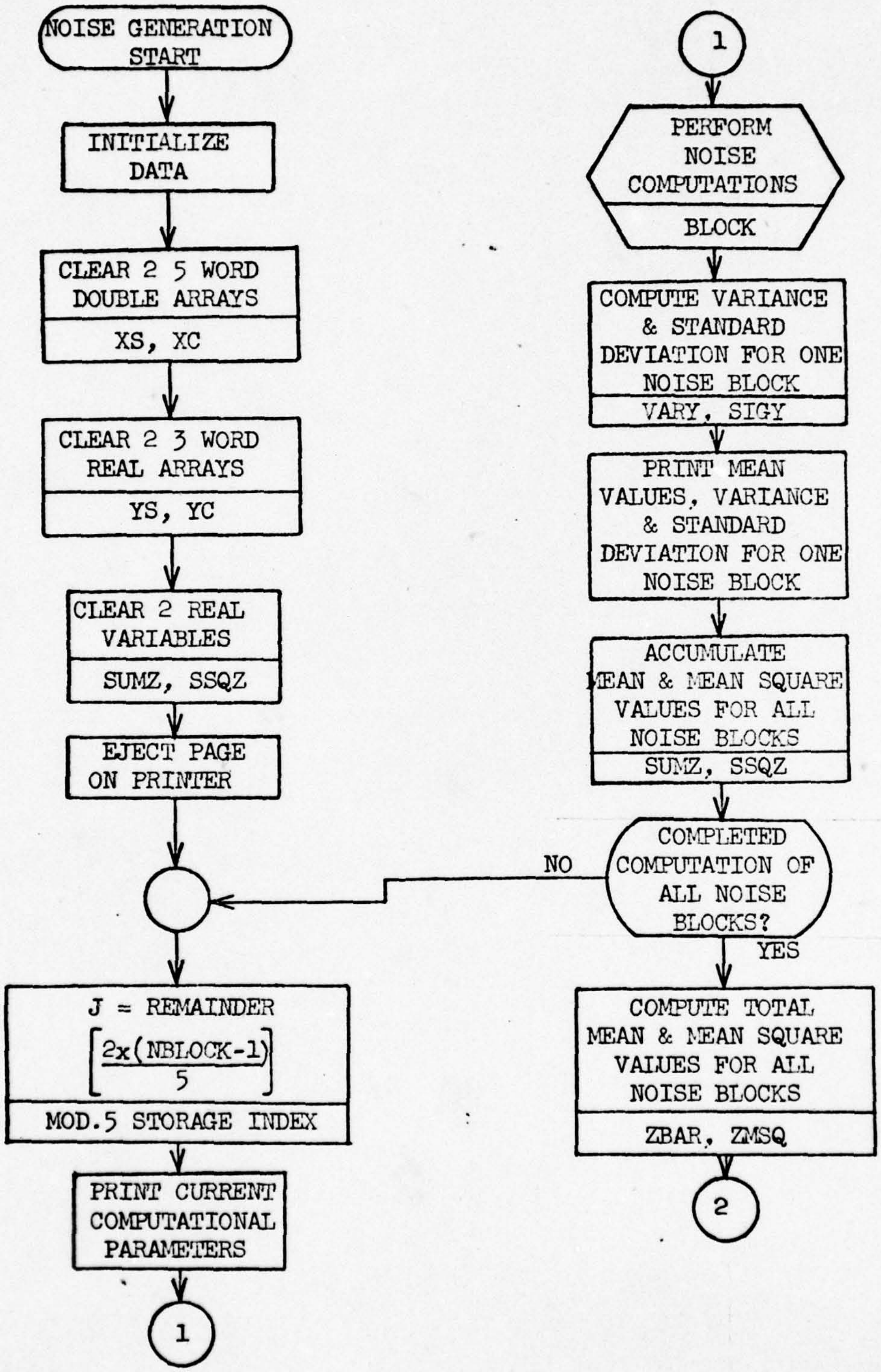
DETAILED FLOWCHART:

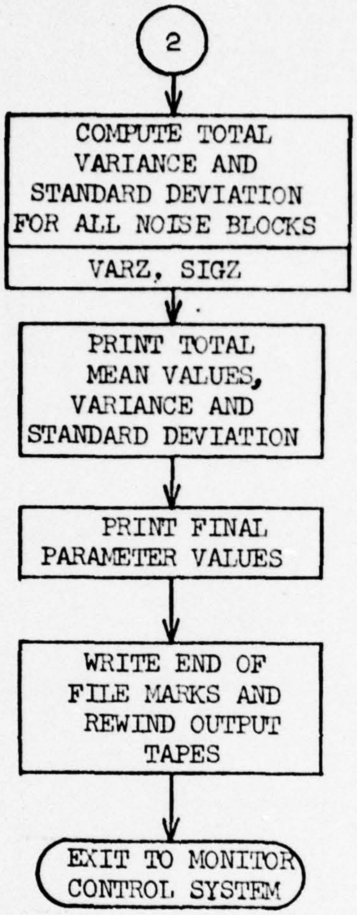
Noise Subroutine

WAVE PERIOD PROCESSOR NOISE GENERATION

PROGRAM FLOWCHART

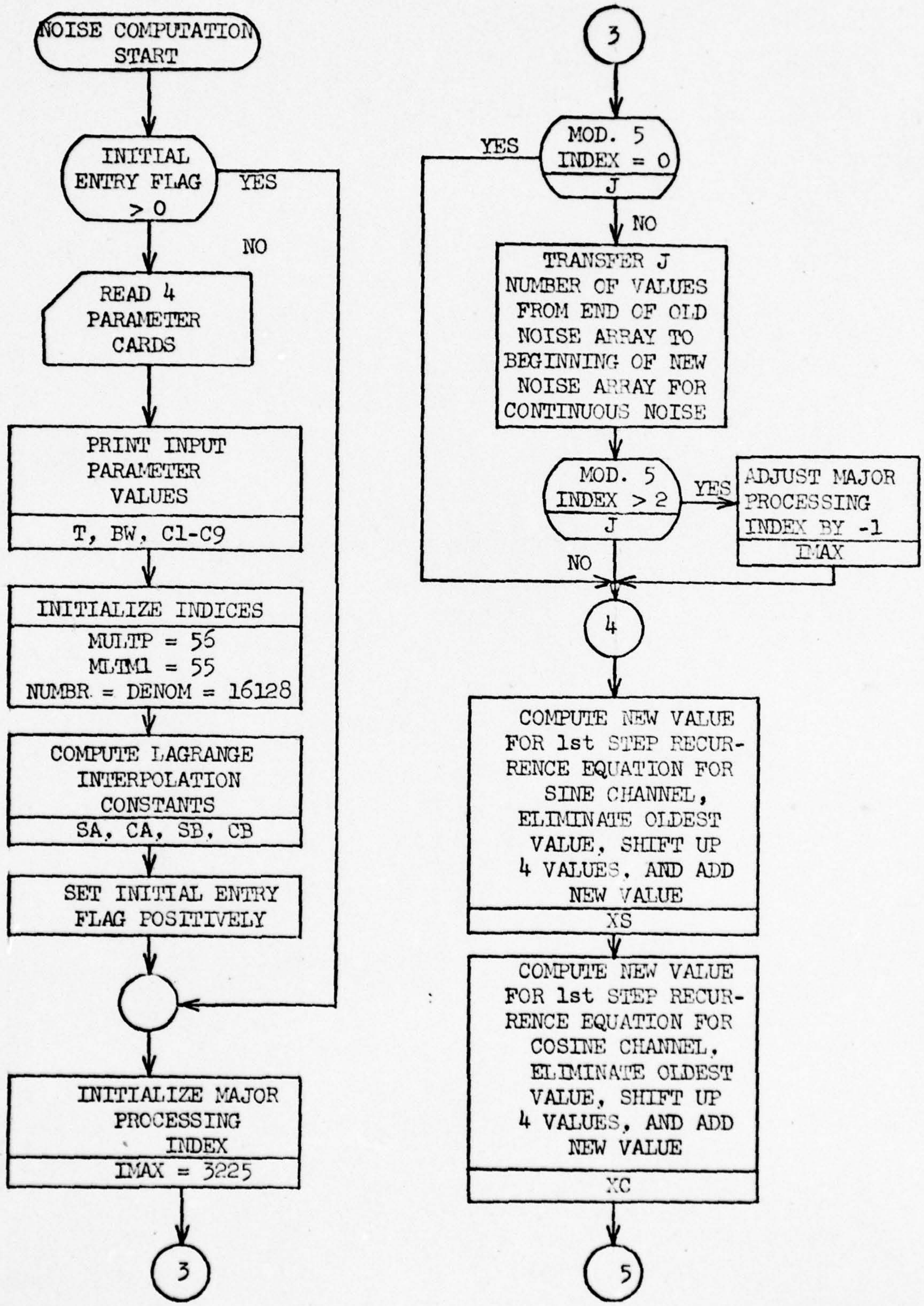
CONTROL ROUTINE: KAVEE

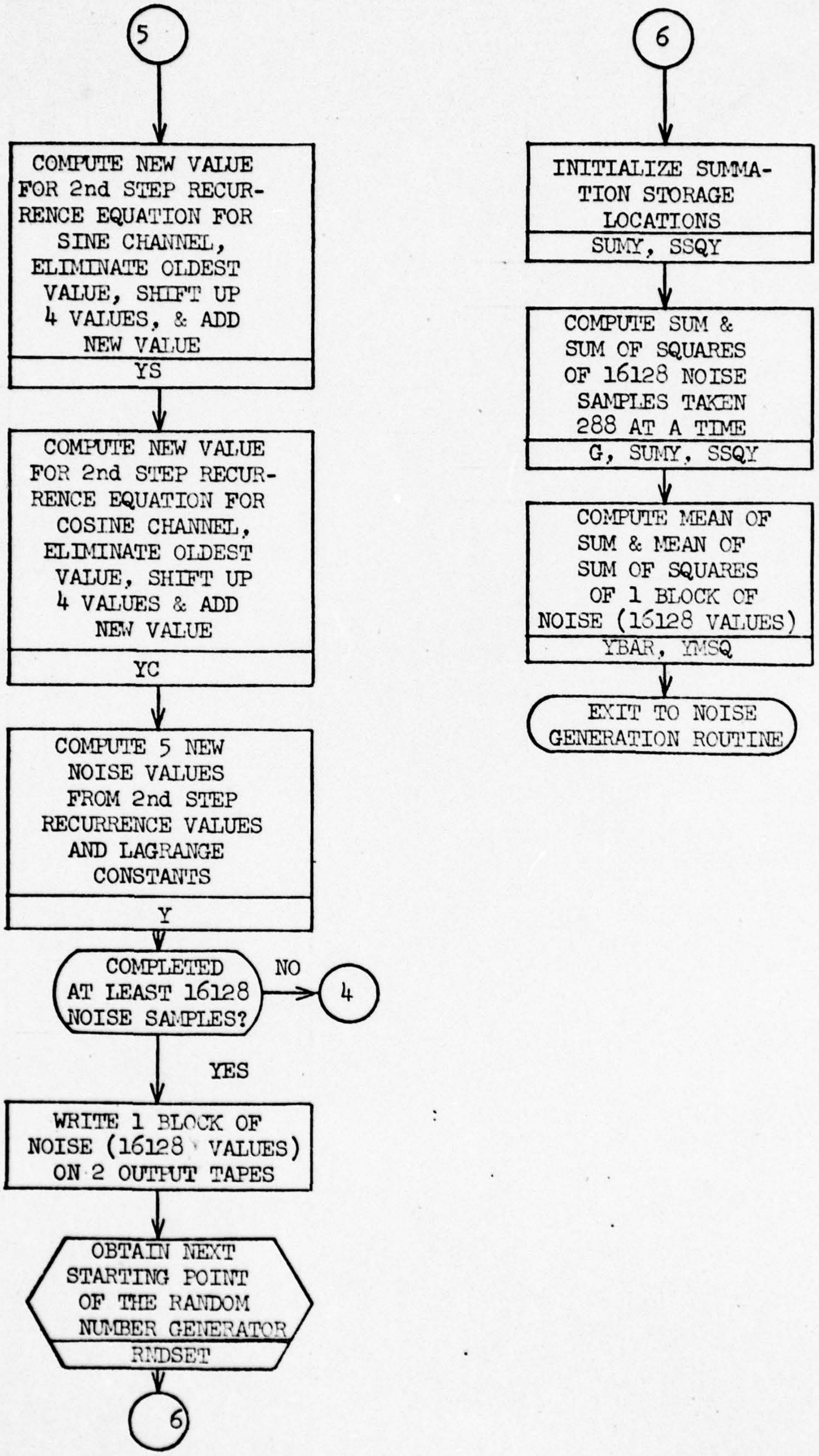

NOISE GENERATION START
INITIALIZE DATA
CLEAR 2 5 WORD DOUBLE ARRAYS
XS, XC
CLEAR 2 3 WORD REAL ARRAYS
YS, YC
CLEAR 2 REAL VARIABLES
SUMZ, SSQZ
EJECT PAGE ON PRINTER
J = REMAINDER
$\left[\frac{2x(NBLOCK-1)}{5}\right]$
MOD.5 STORAGE INDEX
PRINT CURRENT COMPUTATIONAL PARAMETERS
1
1
PERFORM NOISE COMPUTATIONS
BLOCK
COMPUTE VARIANCE & STANDARD DEVIATION FOR ONE NOISE BLOCK
VARY, SIGY
PRINT MEAN VALUES, VARIANCE & STANDARD DEVIATION FOR ONE NOISE BLOCK
ACCUMULATE MEAN & MEAN SQUARE VALUES FOR ALL NOISE BLOCKS
SUMZ, SSQZ
COMPLETED COMPUTATION OF ALL NOISE BLOCKS?
NO
YES
COMPUTE TOTAL MEAN & MEAN SQUARE VALUES FOR ALL NOISE BLOCKS
ZBAR, ZMSQ
2


2
COMPUTE TOTAL VARIANCE AND STANDARD DEVIATION FOR ALL NOISE BLOCKS
VARZ, SIGZ
PRINT TOTAL MEAN VALUES, VARIANCE AND STANDARD DEVIATION
PRINT FINAL PARAMETER VALUES
WRITE END OF FILE MARKS AND REWIND OUTPUT TAPES
EXIT TO MONITOR CONTROL SYSTEM

PROGRAM FLOWCHART

COMPUTATIONAL ROUTINE: BLOCK


NOISE COMPUTATION START
INITIAL ENTRY FLAG > 0
YES
NO
READ 4 PARAMETER CARDS
PRINT INPUT PARAMETER VALUES
T, BW, C1-C9
INITIALIZE INDICES
MULTP = 56
MLTM1 = 55
NUMBR = DENOM = 16128
COMPUTE LAGRANGE INTERPOLATION CONSTANTS
SA, CA, SB, CB
SET INITIAL ENTRY FLAG POSITIVELY
INITIALIZE MAJOR PROCESSING INDEX
IMAX = 3225
3
3
MOD. 5 INDEX = 0
J
YES
NO
TRANSFER J NUMBER OF VALUES FROM END OF OLD NOISE ARRAY TO BEGINNING OF NEW NOISE ARRAY FOR CONTINUOUS NOISE
MOD. 5 INDEX > 2
J
YES
NO
ADJUST MAJOR PROCESSING INDEX BY -1
IMAX
4
COMPUTE NEW VALUE FOR 1st STEP RECURRENCE EQUATION FOR SINE CHANNEL, ELIMINATE OLDEST VALUE, SHIFT UP 4 VALUES, AND ADD NEW VALUE
XS
COMPUTE NEW VALUE FOR 1st STEP RECURRENCE EQUATION FOR COSINE CHANNEL, ELIMINATE OLDEST VALUE, SHIFT UP 4 VALUES, AND ADD NEW VALUE
XC
5


5
COMPUTE NEW VALUE FOR 2nd STEP RECURRENCE EQUATION FOR SINE CHANNEL, ELIMINATE OLDEST VALUE, SHIFT UP 4 VALUES, & ADD NEW VALUE
YS
COMPUTE NEW VALUE FOR 2nd STEP RECURRENCE EQUATION FOR COSINE CHANNEL, ELIMINATE OLDEST VALUE, SHIFT UP 4 VALUES & ADD NEW VALUE
YC
COMPUTE 5 NEW NOISE VALUES FROM 2nd STEP RECURRENCE VALUES AND LAGRANGE CONSTANTS
Y
COMPLETED AT LEAST 16128 NOISE SAMPLES?
NO
4
YES
WRITE 1 BLOCK OF NOISE (16128 VALUES) ON 2 OUTPUT TAPES
OBTAIN NEXT STARTING POINT OF THE RANDOM NUMBER GENERATOR
RNDSET
6
6
INITIALIZE SUMMATION STORAGE LOCATIONS
SUMY, SSQY
COMPUTE SUM & SUM OF SQUARES OF 16128 NOISE SAMPLES TAKEN 288 AT A TIME
G, SUMY, SSQY
COMPUTE MEAN OF SUM & MEAN OF SUM OF SQUARES OF 1 BLOCK OF NOISE (16128 VALUES)
YBAR, YMSQ
EXIT TO NOISE GENERATION ROUTINE

FUNCTIONAL FLOWCHART:

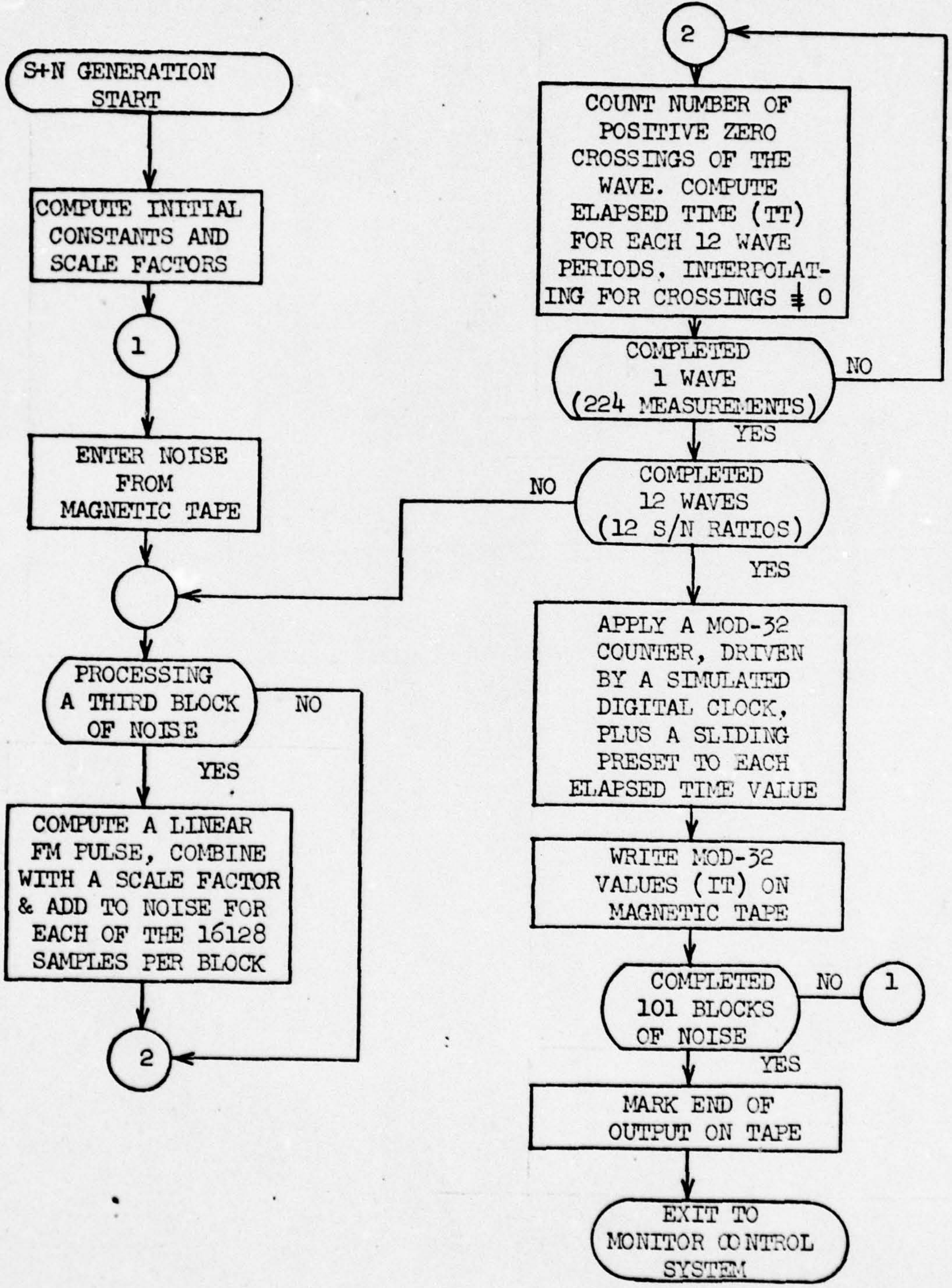
Wave Subroutine

WAVE PERIOD PROCESSOR

SIGNAL PLUS NOISE GENERATION

FUNCTIONAL FLOWCHART

ROUTINE: WAVE


S+N GENERATION START
COMPUTE INITIAL CONSTANTS AND SCALE FACTORS
1
ENTER NOISE FROM MAGNETIC TAPE
PROCESSING A THIRD BLOCK OF NOISE
NO
YES
COMPUTE A LINEAR FM PULSE, COMBINE WITH A SCALE FACTOR & ADD TO NOISE FOR EACH OF THE 16128 SAMPLES PER BLOCK
2
2
COUNT NUMBER OF POSITIVE ZERO CROSSINGS OF THE WAVE. COMPUTE ELAPSED TIME (TT) FOR EACH 12 WAVE PERIODS, INTERPOLATING FOR CROSSINGS ≢ 0
COMPLETED 1 WAVE (224 MEASUREMENTS)
NO
YES
COMPLETED 12 WAVES (12 S/N RATIOS)
NO
YES
APPLY A MOD-32 COUNTER, DRIVEN BY A SIMULATED DIGITAL CLOCK, PLUS A SLIDING PRESET TO EACH ELAPSED TIME VALUE
WRITE MOD-32 VALUES (IT) ON MAGNETIC TAPE
COMPLETED 101 BLOCKS OF NOISE
NO
1
YES
MARK END OF OUTPUT ON TAPE
EXIT TO MONITOR CONTROL SYSTEM

DETAILED FLOWCHART:

Wave Subroutine

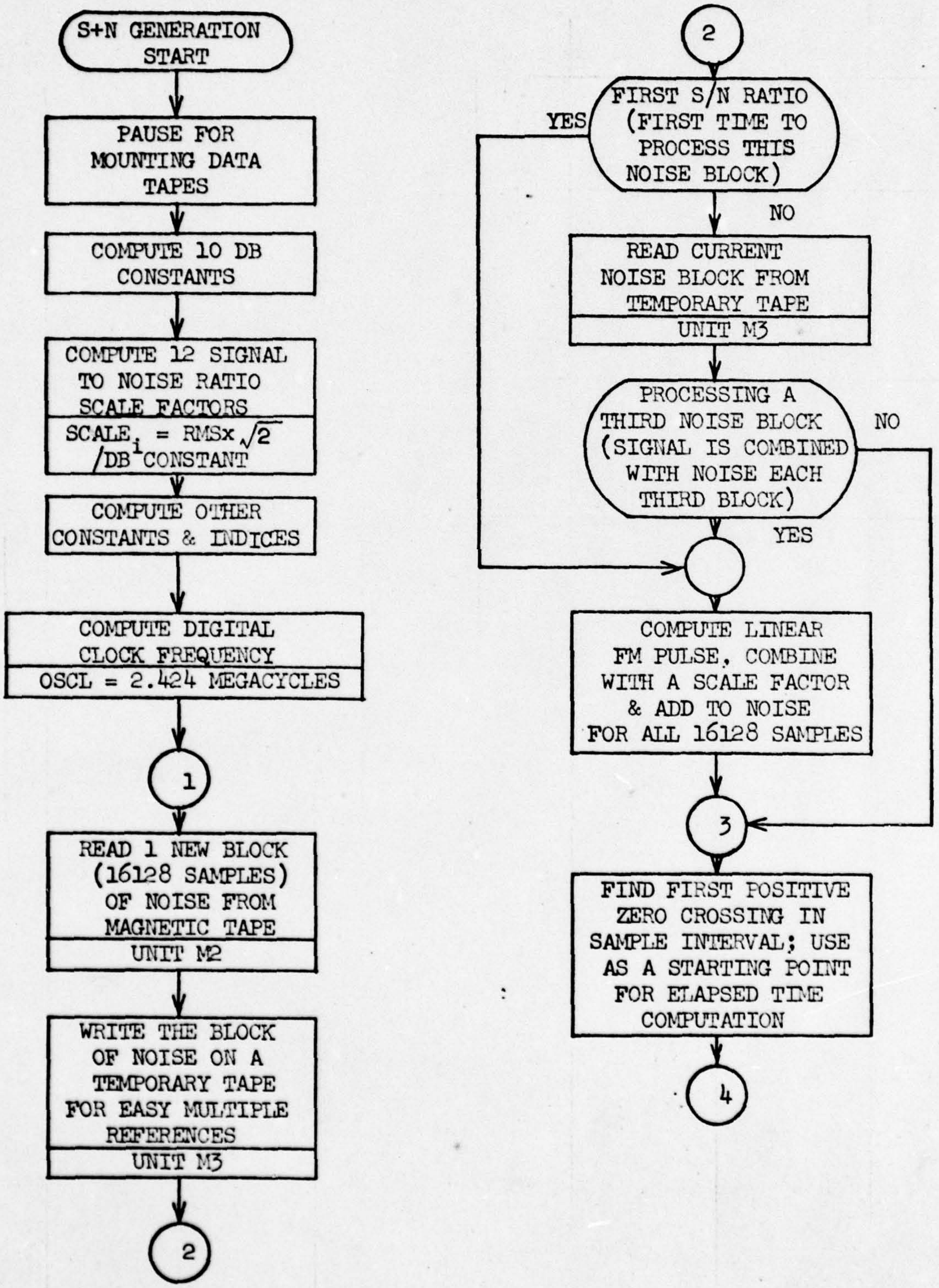
CONFIDENTIAL

WAVE PERIOD PROCESSOR

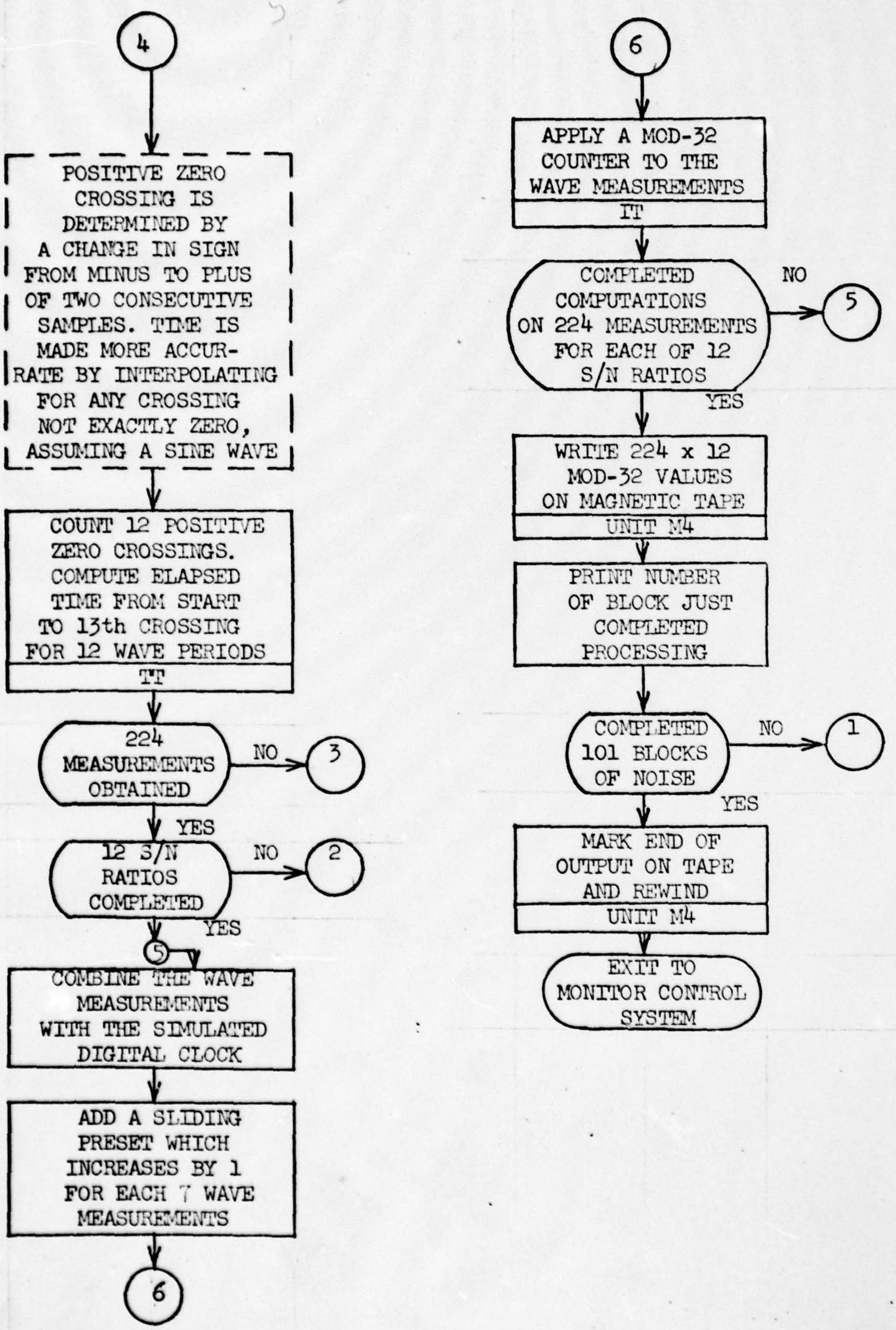
SIGNAL PLUS NOISE GENERATION

PROGRAM FLOWCHART

ROUTINE: WAVE

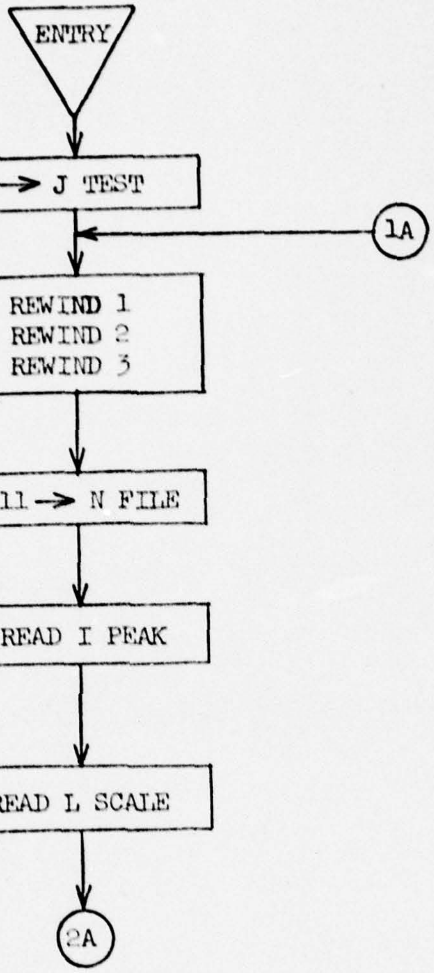

S+N GENERATION START
PAUSE FOR MOUNTING DATA TAPES
COMPUTE 10 DB CONSTANTS
COMPUTE 12 SIGNAL TO NOISE RATIO SCALE FACTORS
SCALE$_i$ = RMSx $\sqrt{2}$ /DB^i CONSTANT
COMPUTE OTHER CONSTANTS & INDICES
COMPUTE DIGITAL CLOCK FREQUENCY
OSCL = 2.424 MEGACYCLES
1
READ 1 NEW BLOCK (16128 SAMPLES) OF NOISE FROM MAGNETIC TAPE
UNIT M2
WRITE THE BLOCK OF NOISE ON A TEMPORARY TAPE FOR EASY MULTIPLE REFERENCES
UNIT M3
2
2
FIRST S/N RATIO (FIRST TIME TO PROCESS THIS NOISE BLOCK)
YES
NO
READ CURRENT NOISE BLOCK FROM TEMPORARY TAPE
UNIT M3
PROCESSING A THIRD NOISE BLOCK (SIGNAL IS COMBINED WITH NOISE EACH THIRD BLOCK)
NO
YES
COMPUTE LINEAR FM PULSE, COMBINE WITH A SCALE FACTOR & ADD TO NOISE FOR ALL 16128 SAMPLES
3
FIND FIRST POSITIVE ZERO CROSSING IN SAMPLE INTERVAL; USE AS A STARTING POINT FOR ELAPSED TIME COMPUTATION
4

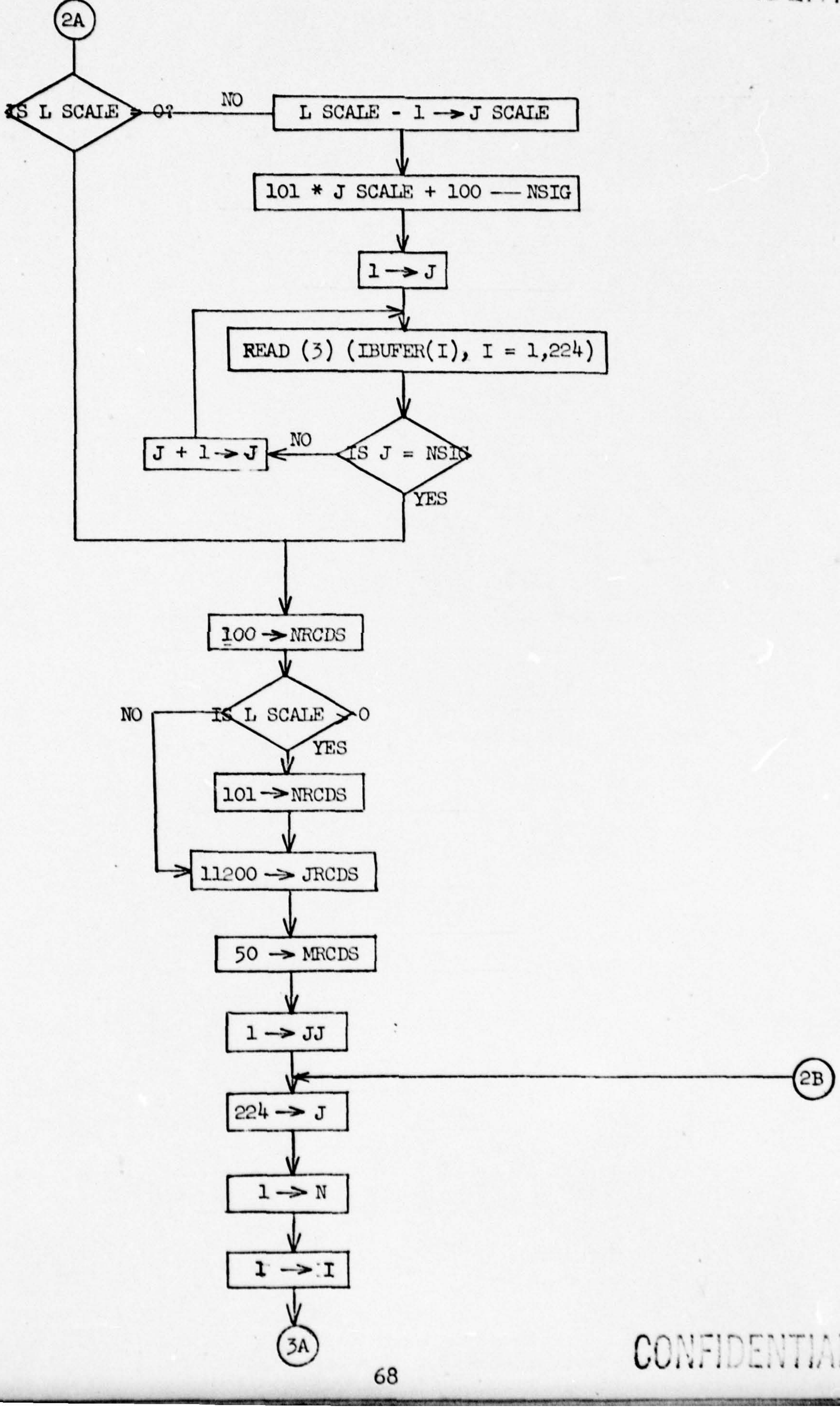
CONFIDENTIAL

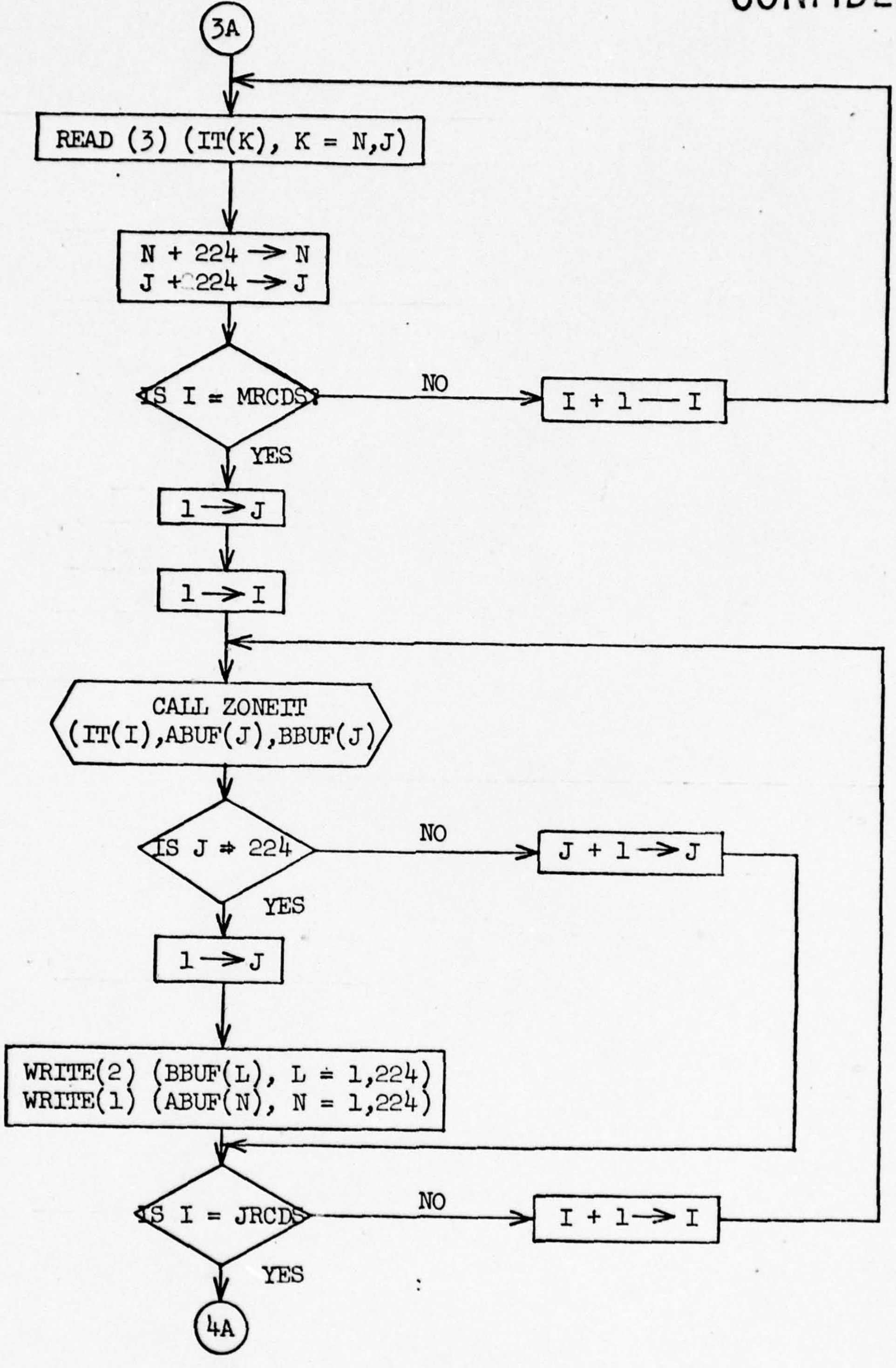

4
POSITIVE ZERO CROSSING IS DETERMINED BY A CHANGE IN SIGN FROM MINUS TO PLUS OF TWO CONSECUTIVE SAMPLES. TIME IS MADE MORE ACCURRATE BY INTERPOLATING FOR ANY CROSSING NOT EXACTLY ZERO, ASSUMING A SINE WAVE
COUNT 12 POSITIVE ZERO CROSSINGS. COMPUTE ELAPSED TIME FROM START TO 13th CROSSING FOR 12 WAVE PERIODS
TT
224 MEASUREMENTS OBTAINED
NO
3
YES
12 S/N RATIOS COMPLETED
NO
2
YES
5
COMBINE THE WAVE MEASUREMENTS WITH THE SIMULATED DIGITAL CLOCK
ADD A SLIDING PRESET WHICH INCREASES BY 1 FOR EACH 7 WAVE MEASUREMENTS
6
6
APPLY A MOD-32 COUNTER TO THE WAVE MEASUREMENTS
IT
COMPLETED COMPUTATIONS ON 224 MEASUREMENTS FOR EACH OF 12 S/N RATIOS
NO
5
YES
WRITE 224 x 12 MOD-32 VALUES ON MAGNETIC TAPE
UNIT M4
PRINT NUMBER OF BLOCK JUST COMPLETED PROCESSING
COMPLETED 101 BLOCKS OF NOISE
NO
1
YES
MARK END OF OUTPUT ON TAPE AND REWIND
UNIT M4
EXIT TO MONITOR CONTROL SYSTEM

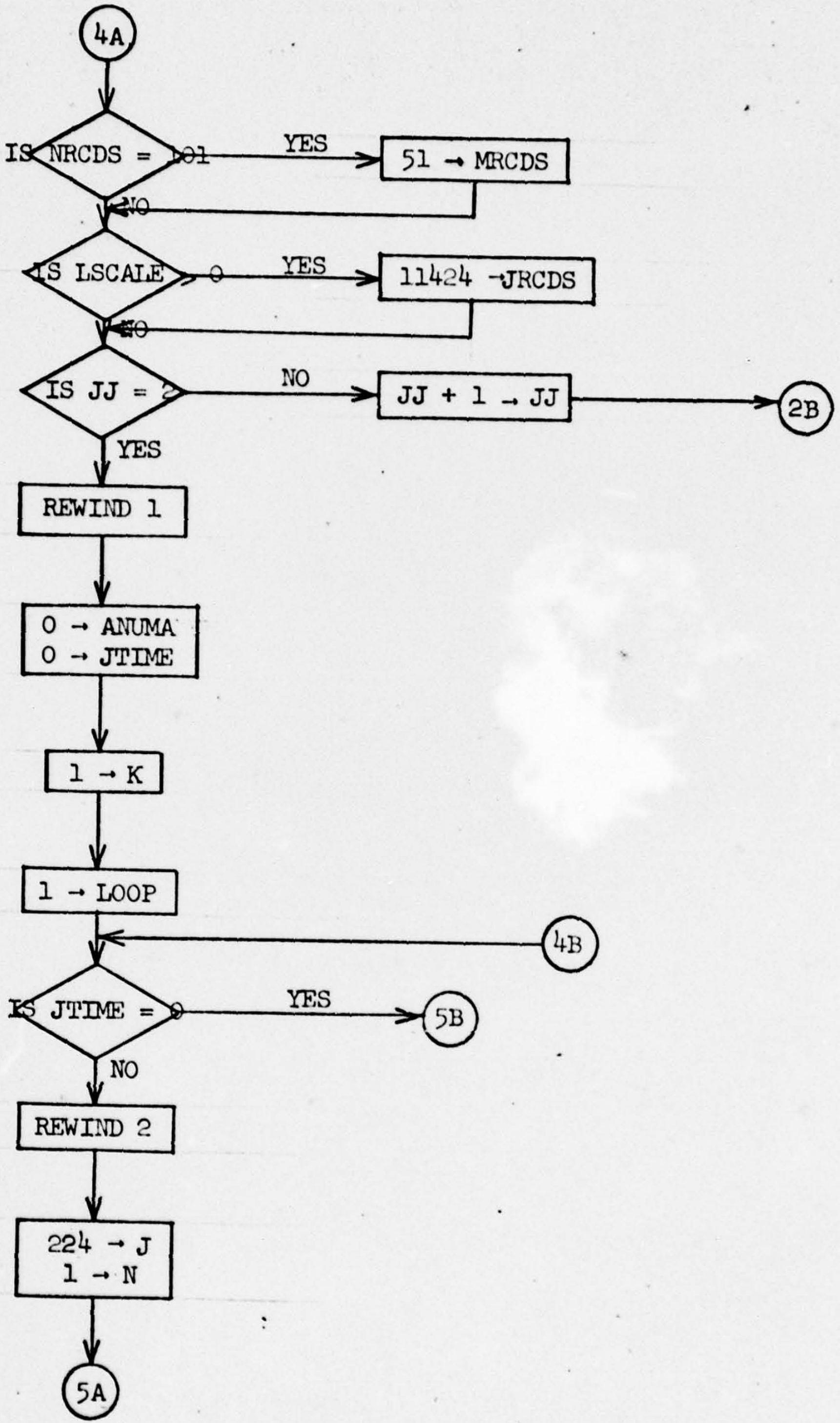
FLOWCHART:

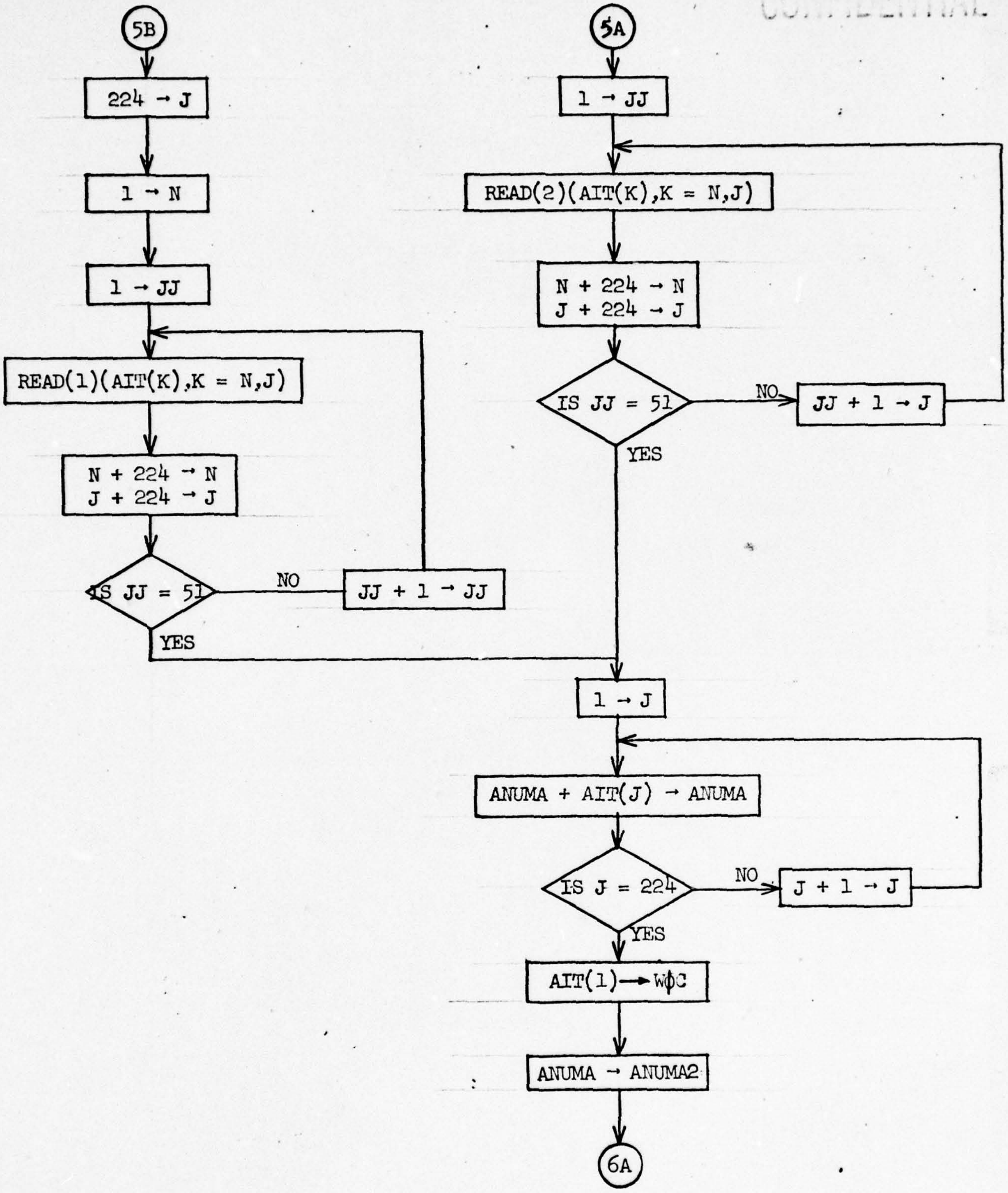
Procs Subroutine

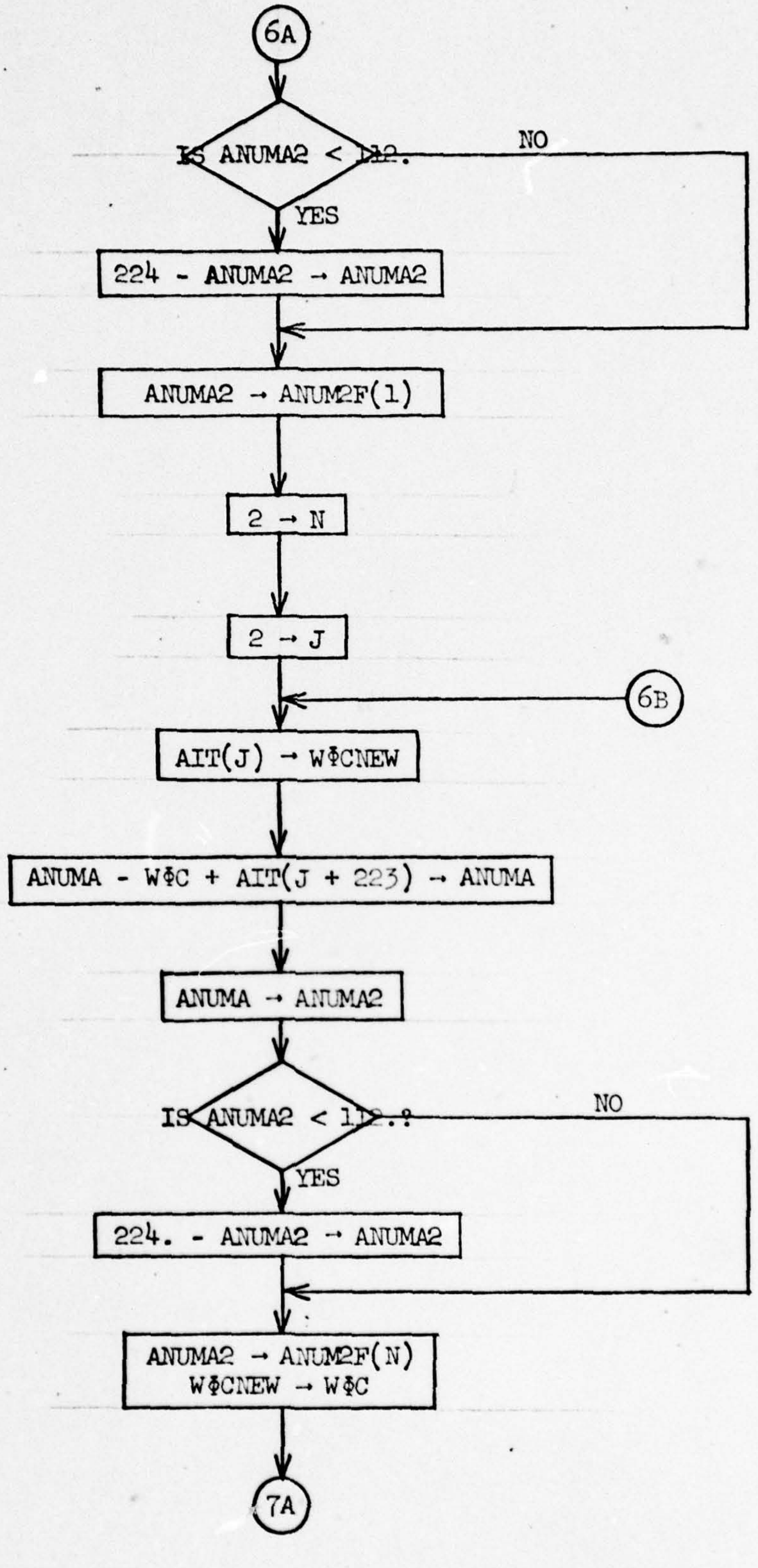

ENTRY
1 → J TEST
1A
REWIND 1
REWIND 2
REWIND 3
1211 → N FILE
READ I PEAK
READ L SCALE
2A

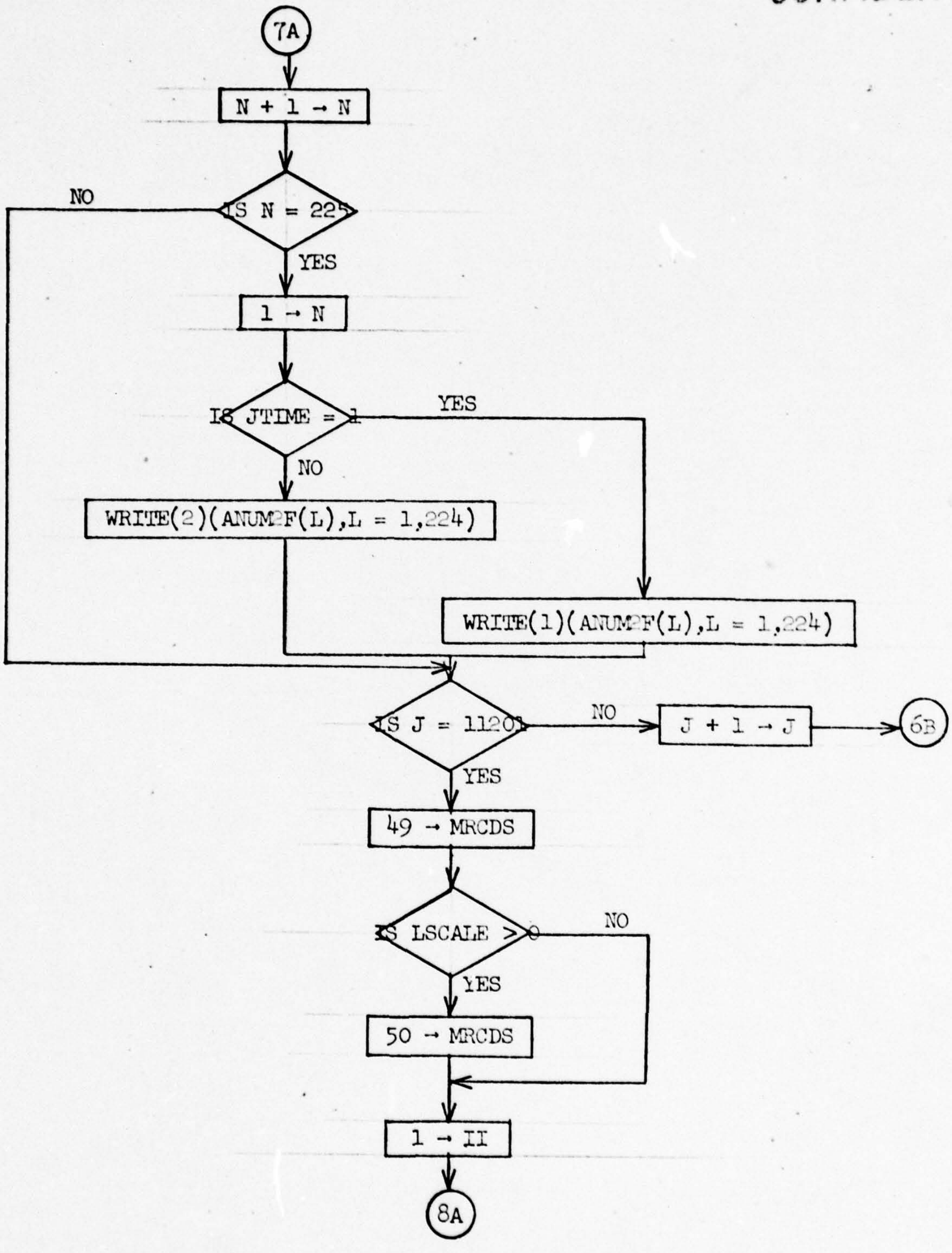

2A
IS L SCALE > 0?
NO
L SCALE - 1 → J SCALE
101 * J SCALE + 100 —— NSIG
1 → J
READ (3) (IBUFER(I), I = 1,224)
IS J = NSIG
NO
J + 1 → J
YES
100 → NRCDS
IS L SCALE > 0
NO
YES
101 → NRCDS
11200 → JRCDS
50 → MRCDS
1 → JJ
2B
224 → J
1 → N
1 → I
3A

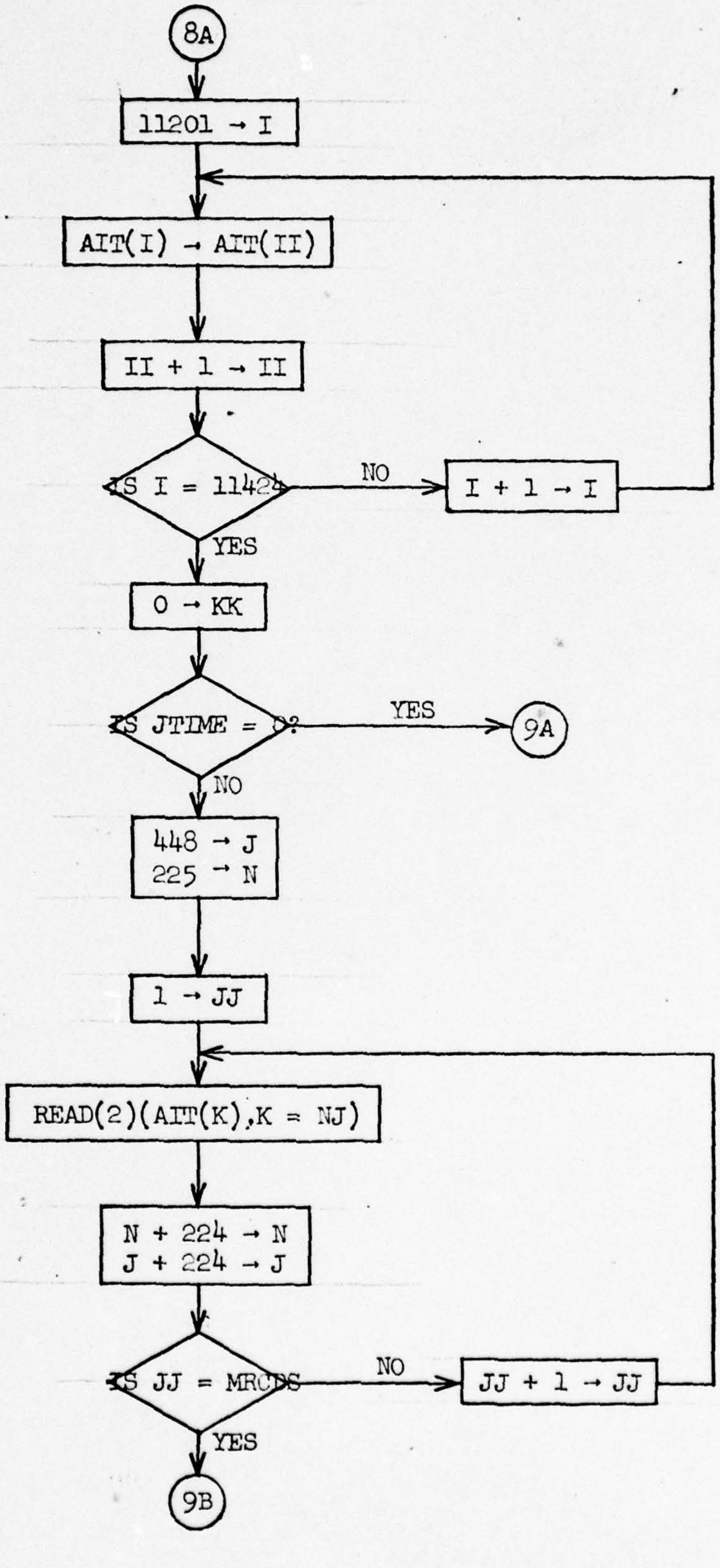

3A
READ (3) (IT(K), K = N,J)
N + 224 → N
J + 224 → J
IS I = MRCDS?
NO
I + 1 — I
YES
1 → J
1 → I
CALL ZONEIT
(IT(I),ABUF(J),BBUF(J))
IS J = 224
NO
J + 1 → J
YES
1 → J
WRITE(2) (BBUF(L), L = 1,224)
WRITE(1) (ABUF(N), N = 1,224)
IS I = JRCDS
NO
I + 1 → I
YES
4A

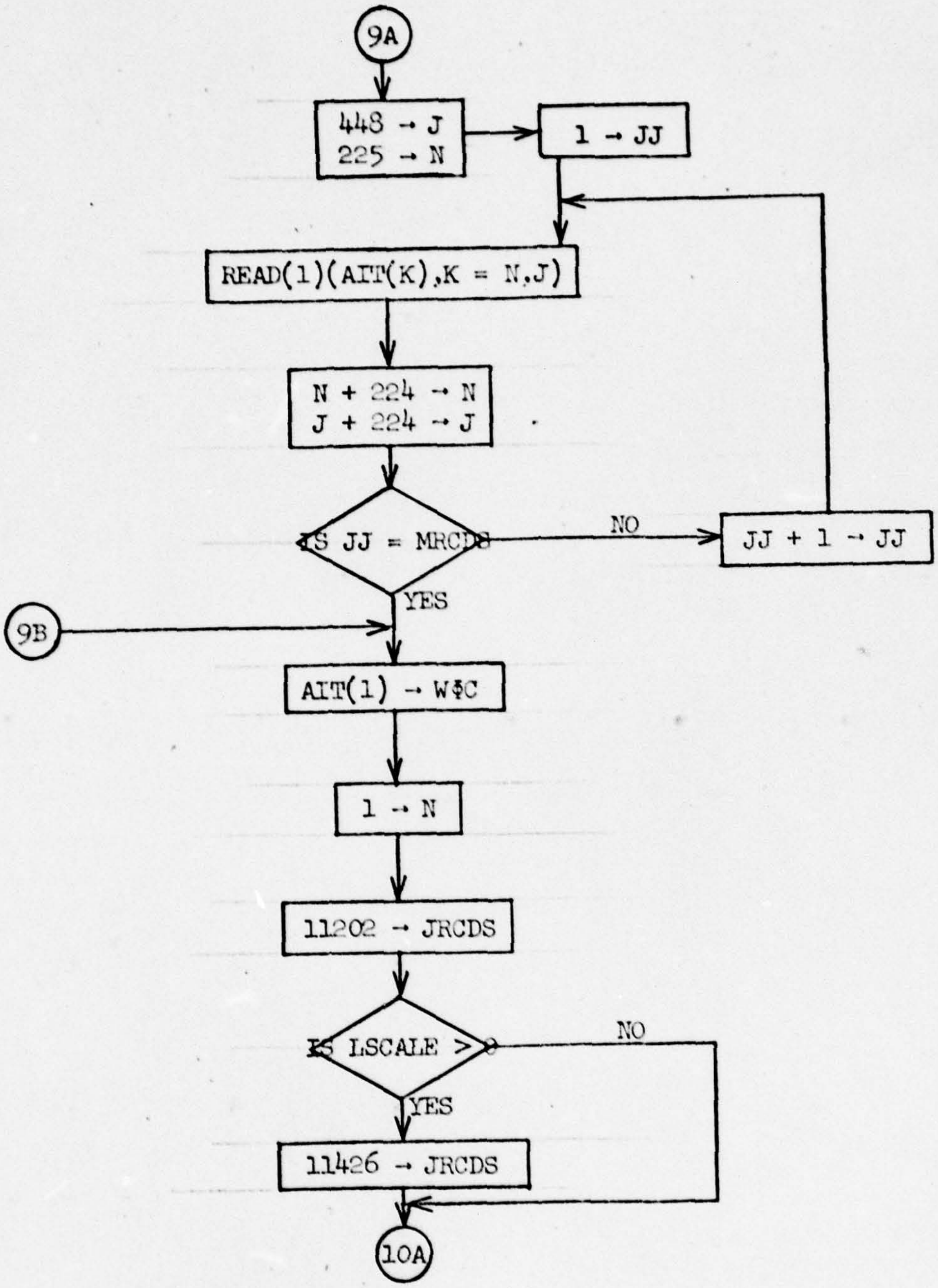

4A
IS NRCDS = 101
YES
51 → MRCDS
NO
IS LSCALE > 0
YES
11424 →JRCDS
NO
IS JJ = 2
NO
JJ + 1 → JJ
2B
YES
REWIND 1
0 → ANUMA
0 → JTIME
1 → K
1 → LOOP
4B
IS JTIME = 0
YES
5B
NO
REWIND 2
224 → J
1 → N
5A

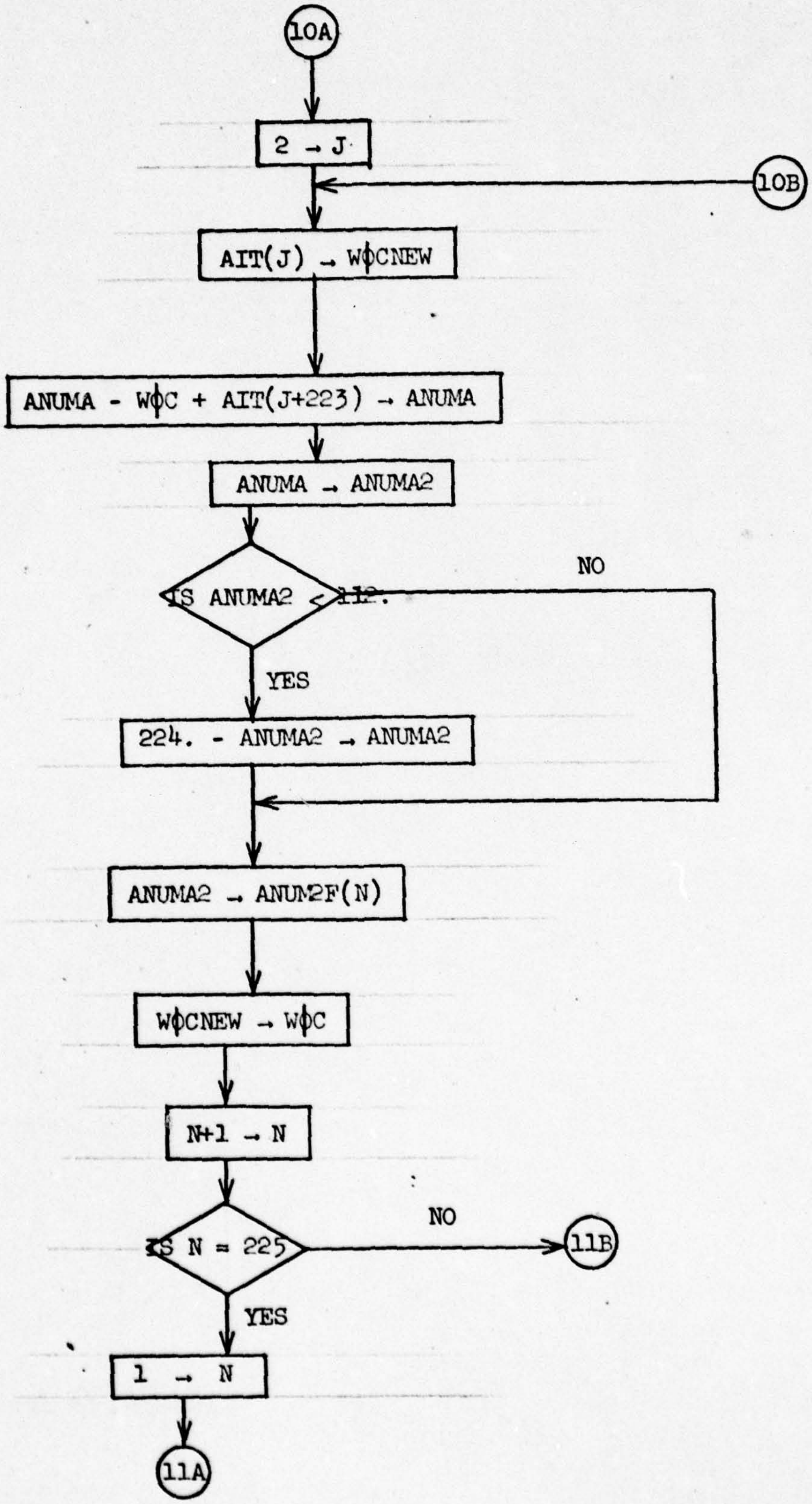

5B
224 → J
1 → N
1 → JJ
READ(1)(AIT(K),K = N,J)
N + 224 → N
J + 224 → J
IS JJ = 51
NO
JJ + 1 → JJ
YES
5A
1 → JJ
READ(2)(AIT(K),K = N,J)
N + 224 → N
J + 224 → J
IS JJ = 51
NO
JJ + 1 → J
YES
1 → J
ANUMA + AIT(J) → ANUMA
IS J = 224
NO
J + 1 → J
YES
AIT(1) → WØC
ANUMA → ANUMA2
6A

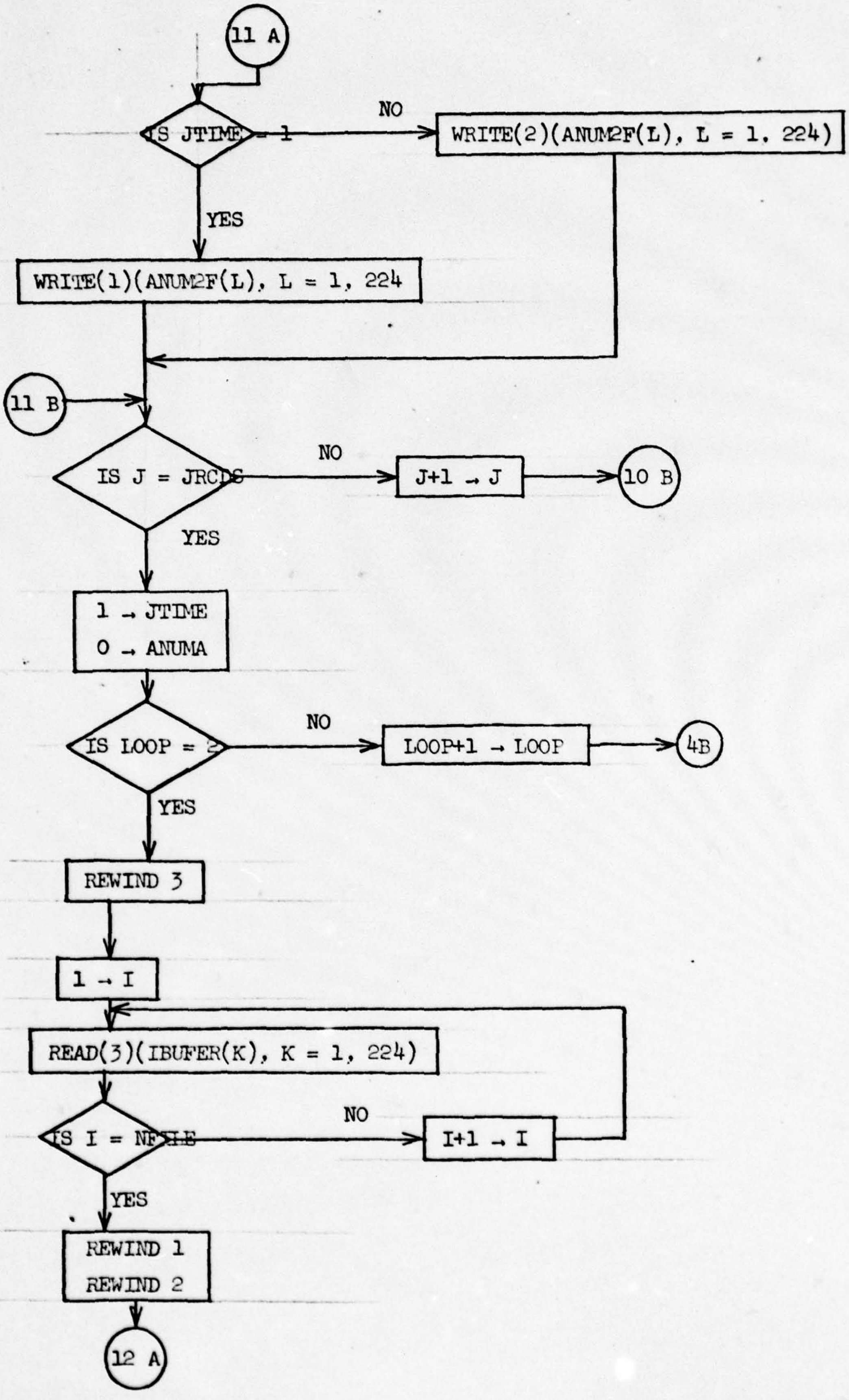

6A
IS ANUMA2 < 112.
NO
YES
224 - ANUMA2 → ANUMA2
ANUMA2 → ANUM2F(1)
2 → N
2 → J
6B
AIT(J) → WΦCNEW
ANUMA - WΦC + AIT(J + 223) → ANUMA
ANUMA → ANUMA2
IS ANUMA2 < 112.?
NO
YES
224. - ANUMA2 → ANUMA2
ANUMA2 → ANUM2F(N)
WΦCNEW → WΦC
7A

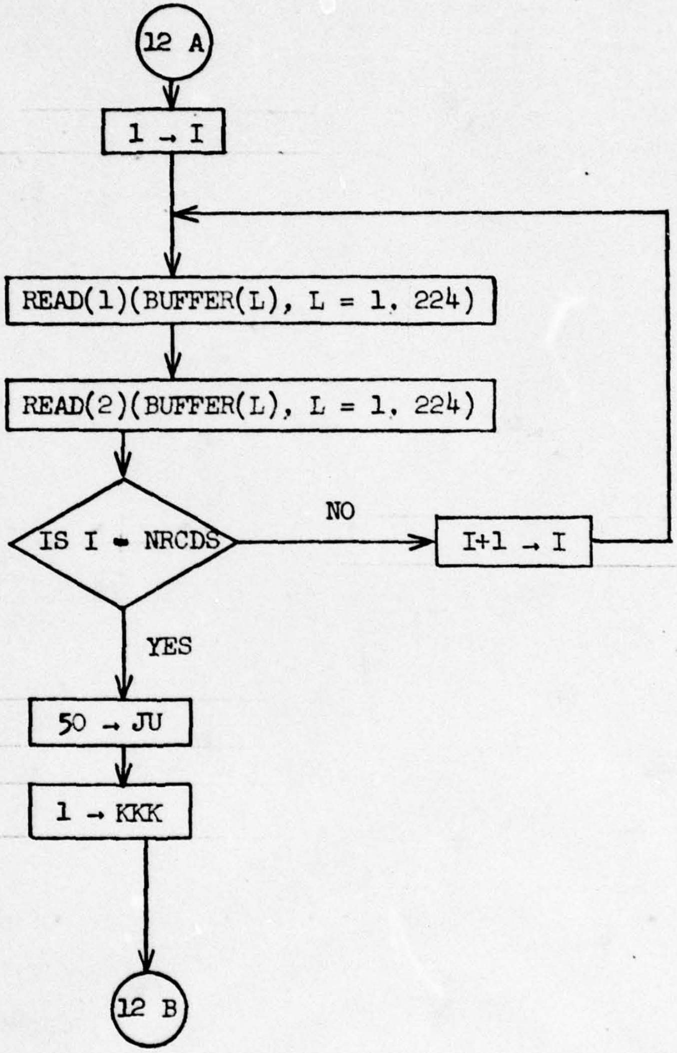

7A
N + 1 → N
IS N = 225
NO
YES
1 → N
IS JTIME = 1
YES
NO
WRITE(2)(ANUM2F(L),L = 1,224)
WRITE(1)(ANUM2F(L),L = 1,224)
IS J = 1120
NO
J + 1 → J
6B
YES
49 → MRCDS
IS LSCALE > 0
NO
YES
50 → MRCDS
1 → II
8A

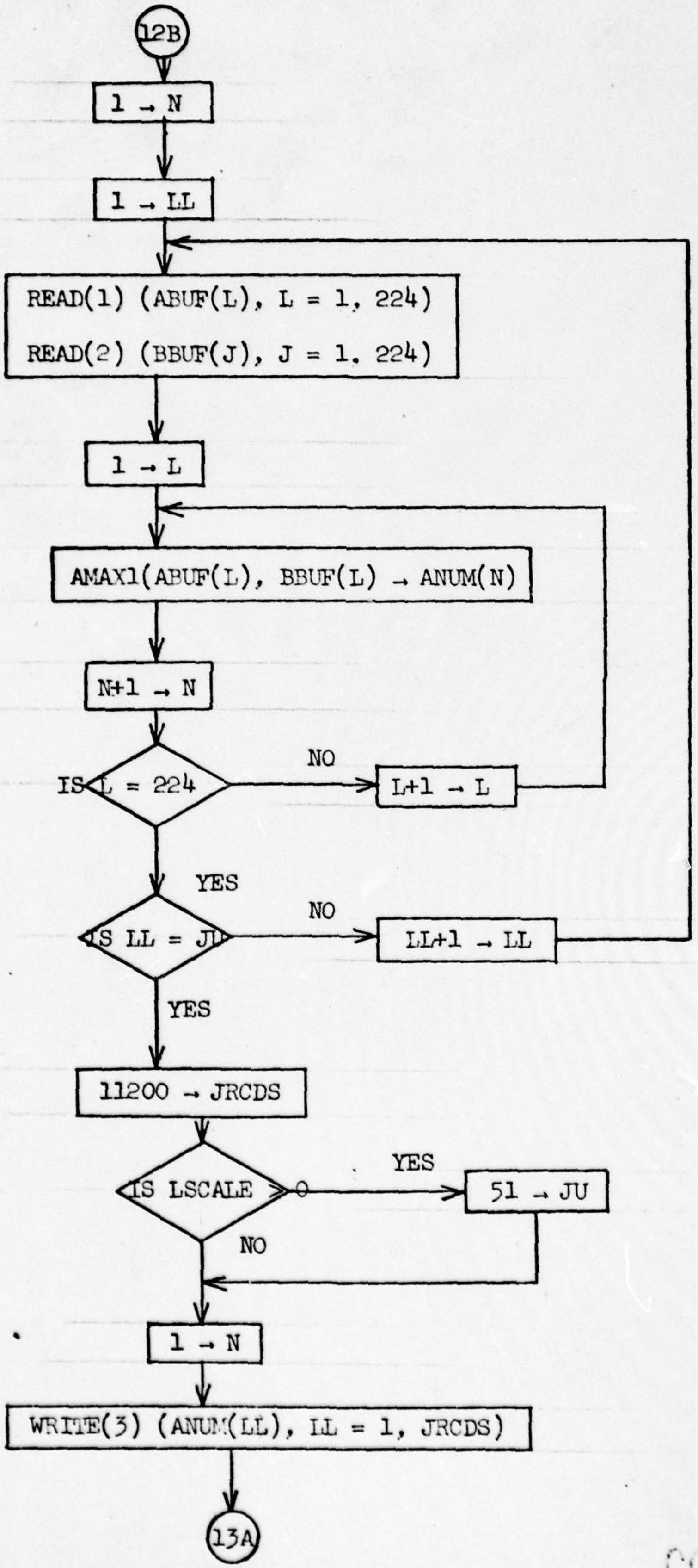

8A
11201 → I
AIT(I) → AIT(II)
II + 1 → II
IS I = 11424
NO
I + 1 → I
YES
0 → KK
IS JTIME = 0?
YES
9A
NO
448 → J
225 → N
1 → JJ
READ(2)(AIT(K),K = NJ)
N + 224 → N
J + 224 → J
IS JJ = MRCDS
NO
JJ + 1 → JJ
YES
9B

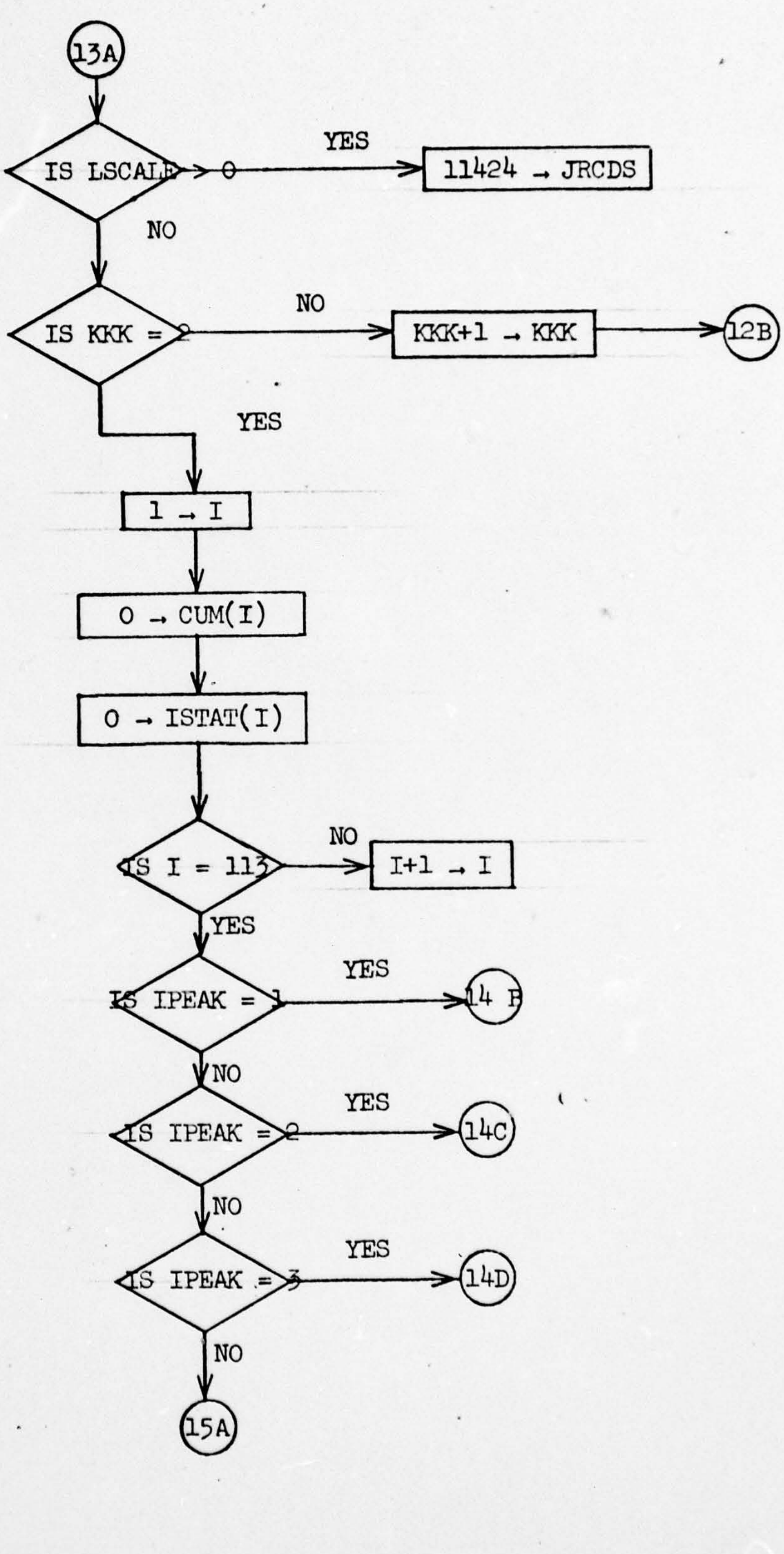

9A
448 → J
225 → N
1 → JJ
READ(1)(AIT(K),K = N,J)
N + 224 → N
J + 224 → J
IS JJ = MRCDS
NO
JJ + 1 → JJ
YES
9B
AIT(1) → W$C
1 → N
11202 → JRCDS
IS LSCALE > 0
NO
YES
11426 → JRCDS
10A

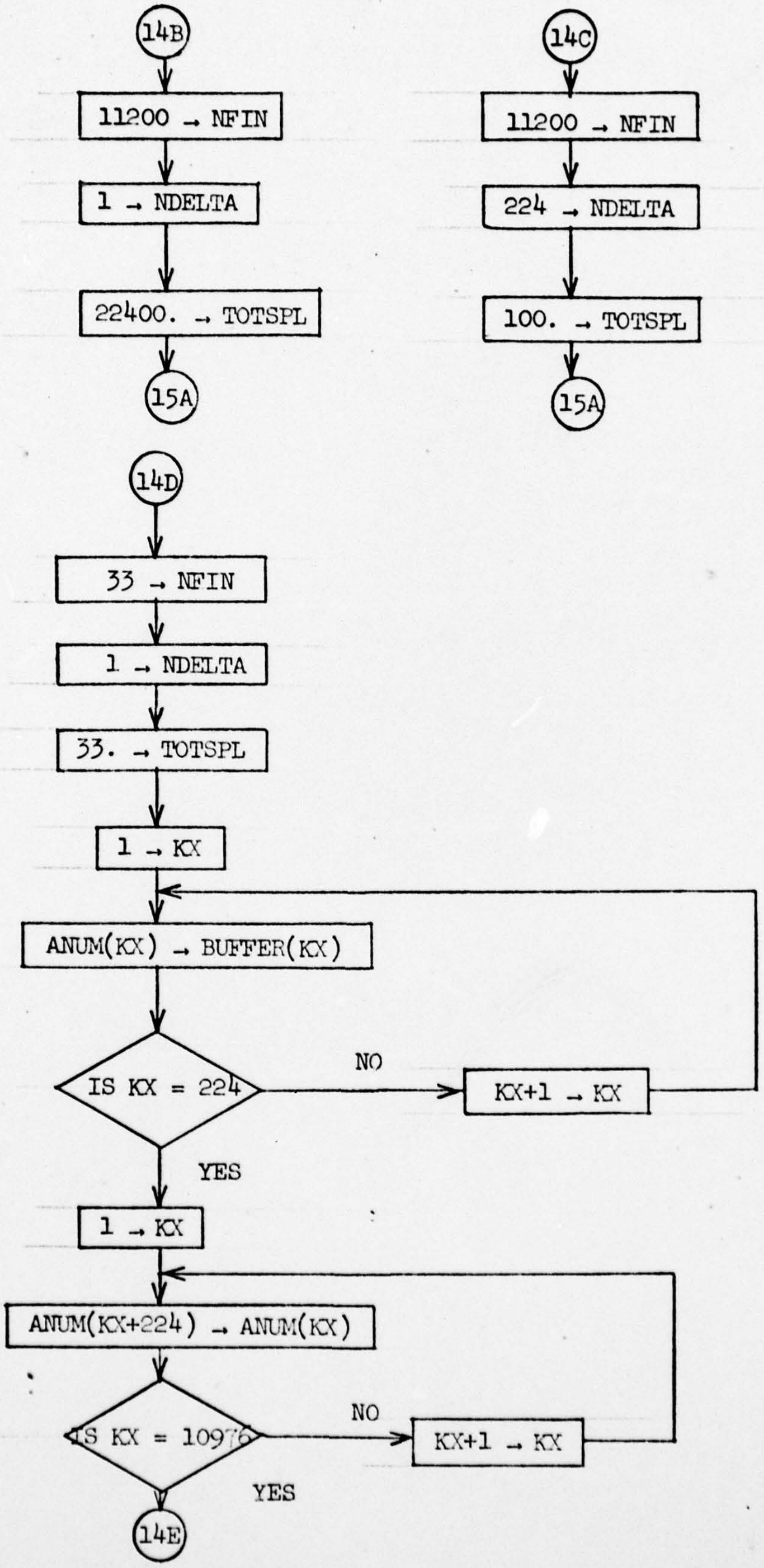

10A
2 → J
10B
AIT(J) → WɸCNEW
ANUMA - WɸC + AIT(J+223) → ANUMA
ANUMA → ANUMA2
IS ANUMA2 < 112.
NO
YES
224. - ANUMA2 → ANUMA2
ANUMA2 → ANUM2F(N)
WɸCNEW → WɸC
N+1 → N
IS N = 225
NO
11B
YES
1 → N
11A

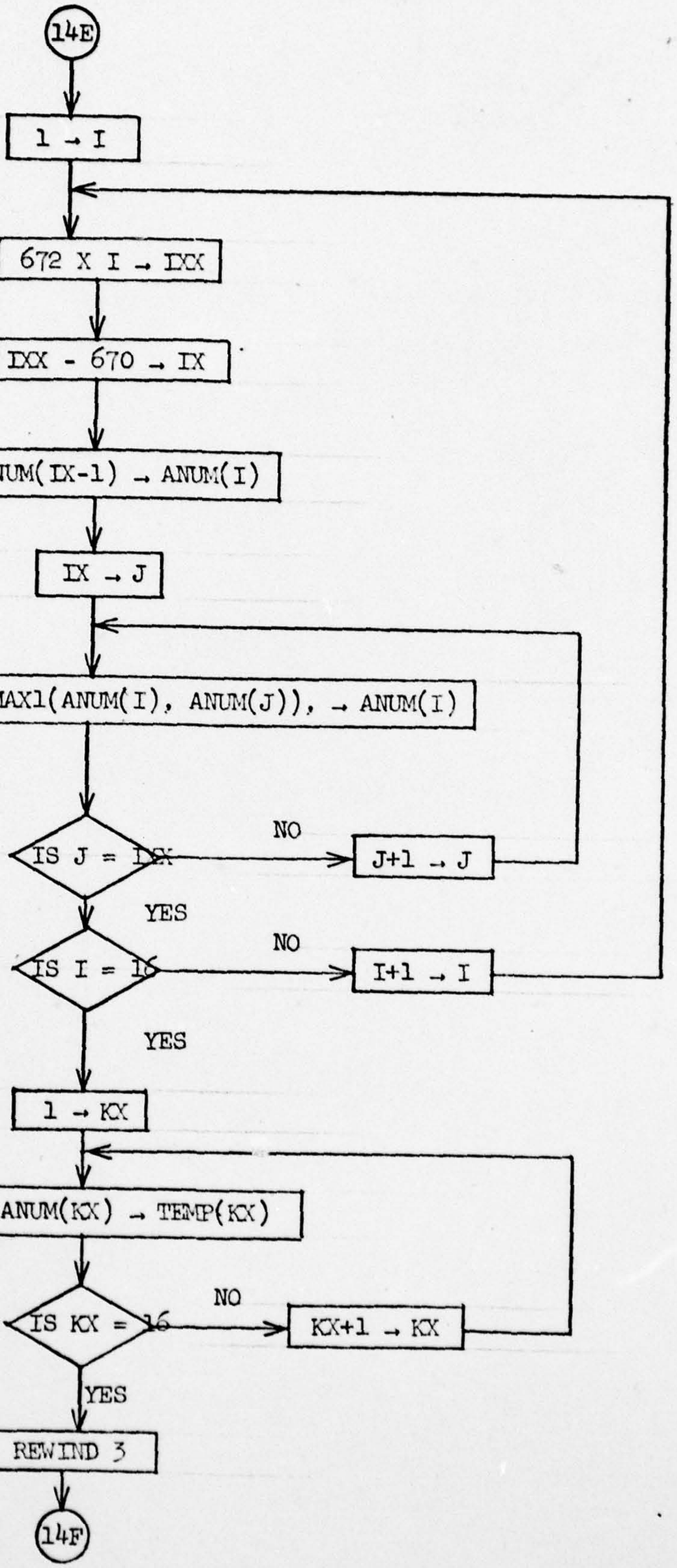

11 A
IS JTIME = 1
NO
WRITE(2)(ANUM2F(L), L = 1, 224)
YES
WRITE(1)(ANUM2F(L), L = 1, 224
11 B
IS J = JRCDS
NO
J+1 → J
10 B
YES
1 → JTIME
0 → ANUMA
IS LOOP = 2
NO
LOOP+1 → LOOP
4B
YES
REWIND 3
1 → I
READ(3)(IBUFER(K), K = 1, 224)
IS I = NFILE
NO
I+1 → I
YES
REWIND 1
REWIND 2
12 A

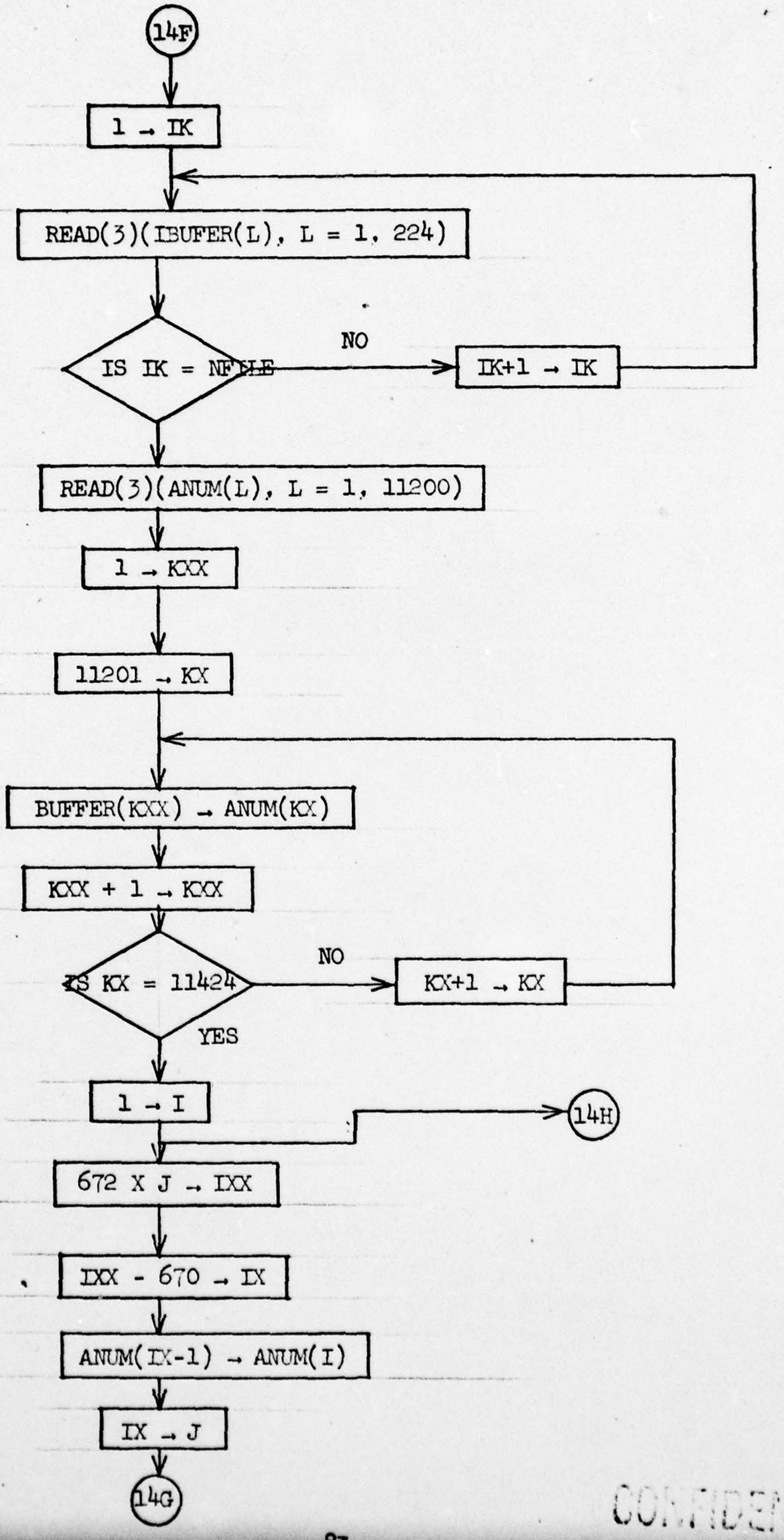

12 A
1 → I
READ(1)(BUFFER(L), L = 1, 224)
READ(2)(BUFFER(L), L = 1, 224)
IS I = NRCDS
NO
I+1 → I
YES
50 → JU
1 → KKK
12 B

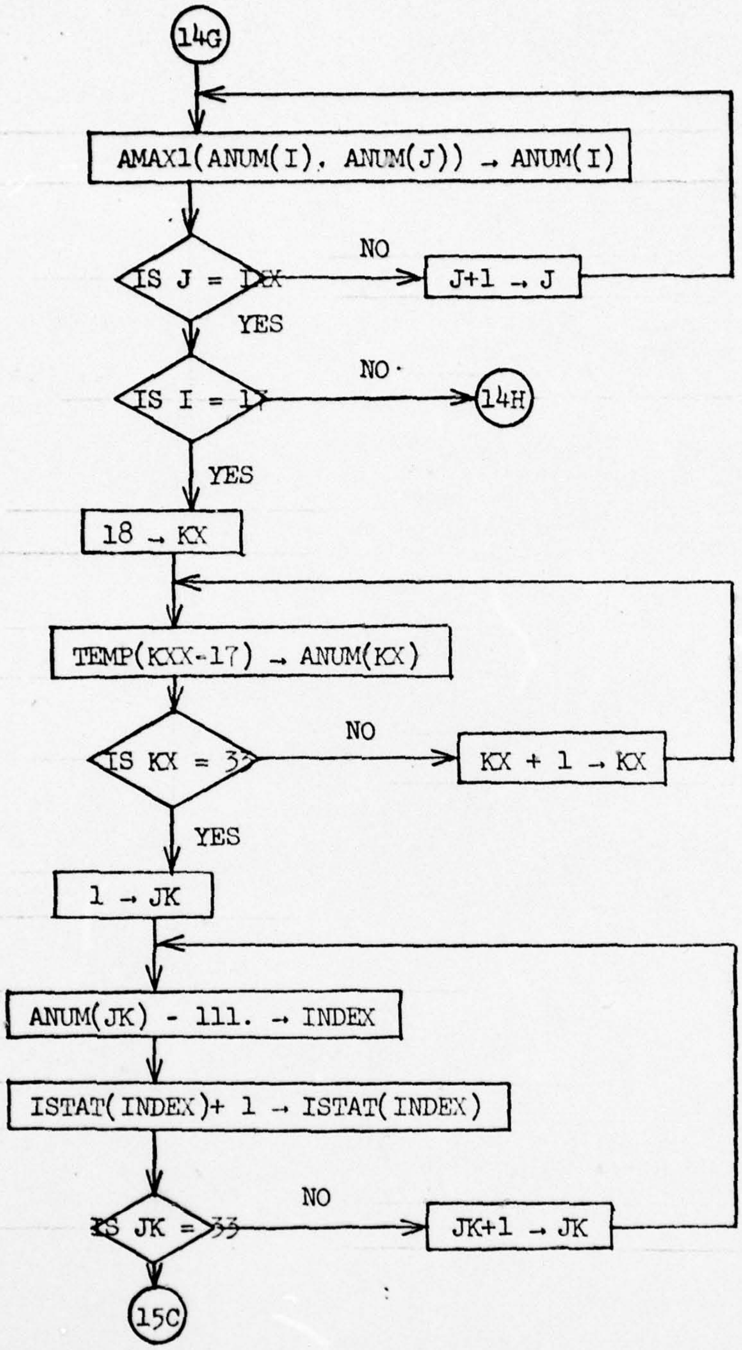

12B
1 → N
1 → LL
READ(1) (ABUF(L), L = 1, 224)
READ(2) (BBUF(J), J = 1, 224)
1 → L
AMAX1(ABUF(L), BBUF(L) → ANUM(N)
N+1 → N
IS L = 224
NO
L+1 → L
YES
IS LL = JU
NO
LL+1 → LL
YES
11200 → JRCDS
IS LSCALE > 0
YES
51 → JU
NO
1 → N
WRITE(3) (ANUM(LL), LL = 1, JRCDS)
13A

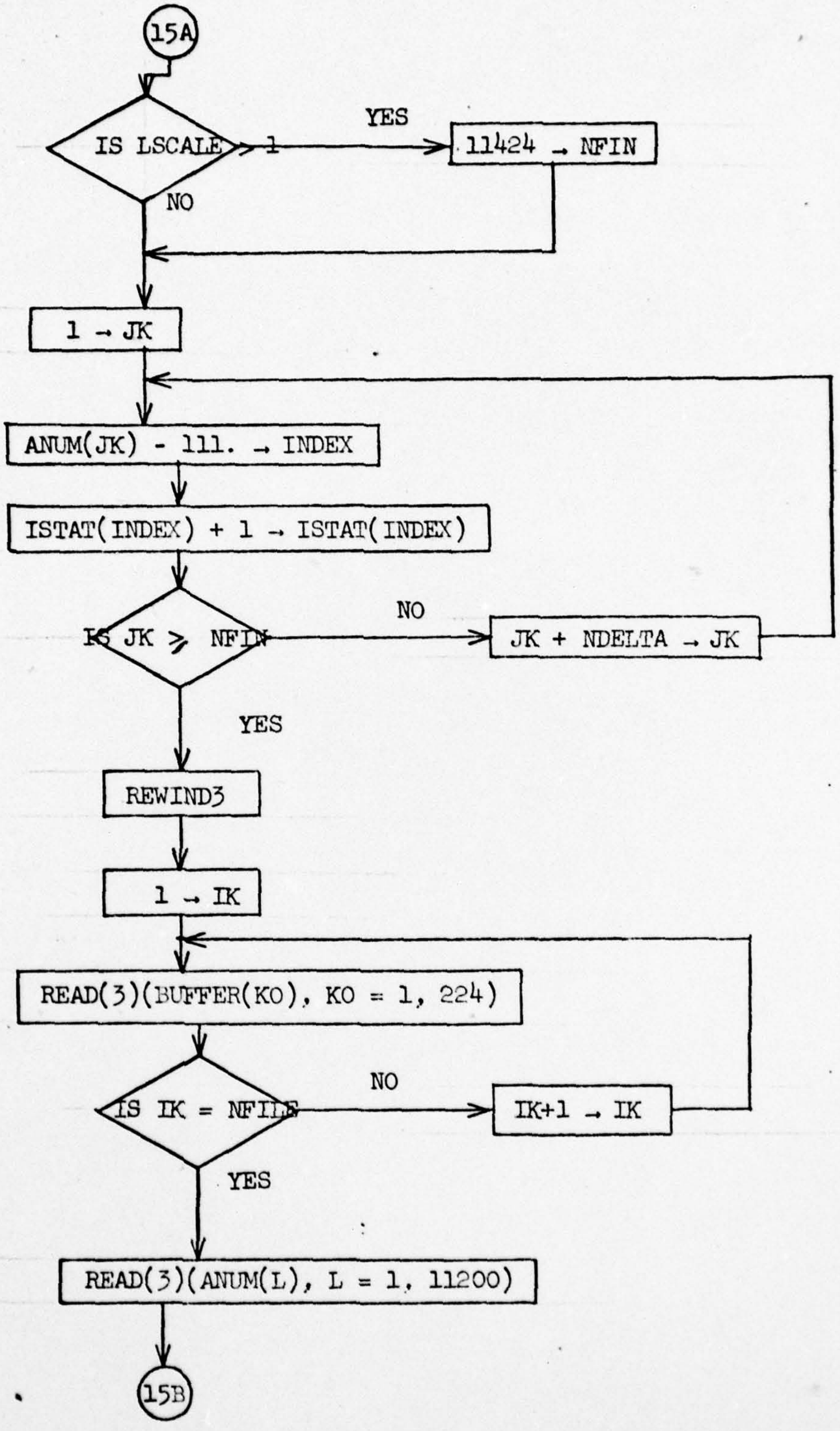

13A
IS LSCALE > 0
YES
11424 → JRCDS
NO
IS KKK = 2
NO
KKK+1 → KKK
12B
YES
1 → I
0 → CUM(I)
0 → ISTAT(I)
IS I = 113
NO
I+1 → I
YES
IS IPEAK = 1
YES
14 B
NO
IS IPEAK = 2
YES
14C
NO
IS IPEAK = 3
YES
14D
NO
15A

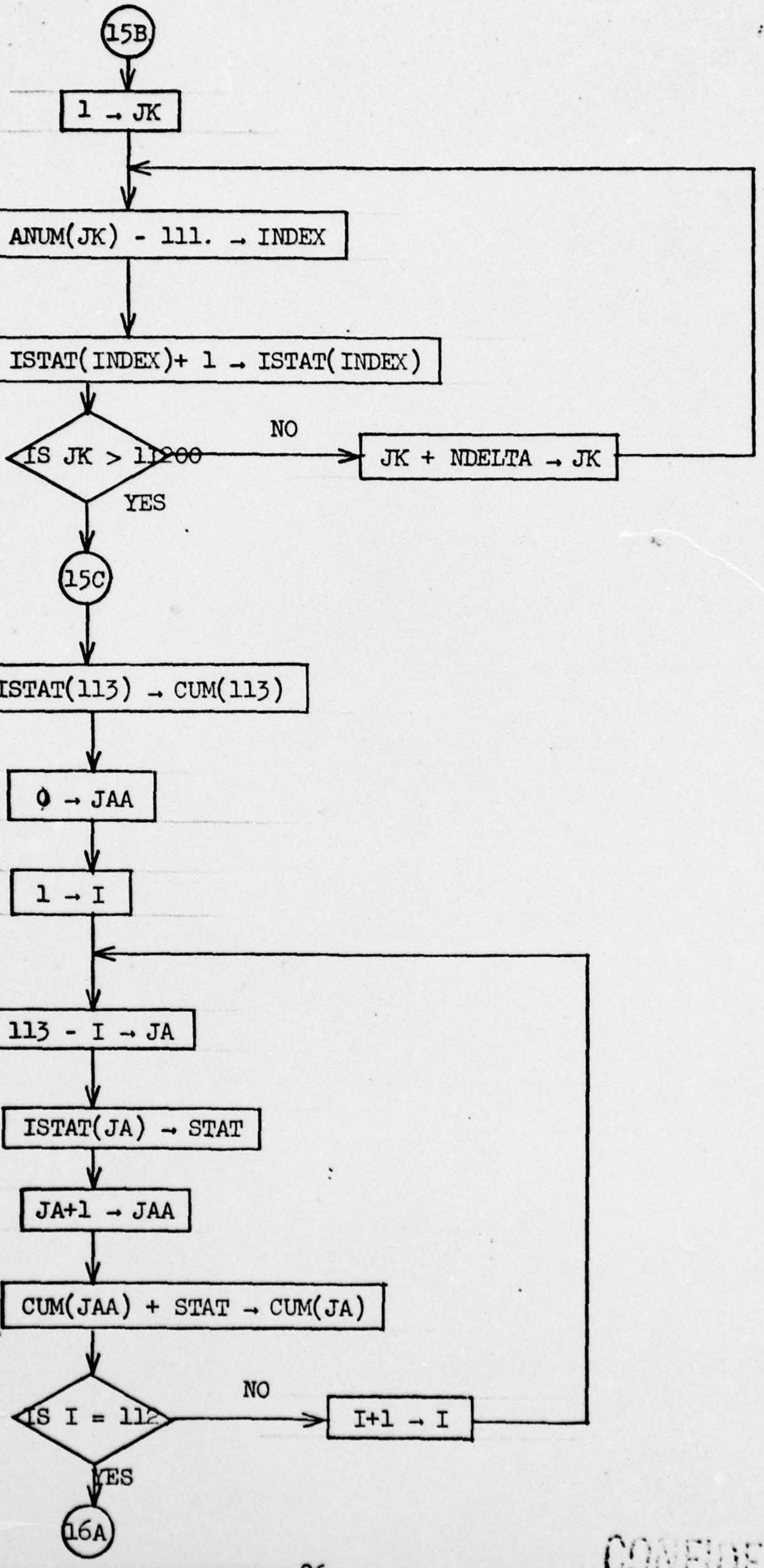

14B
11200 → NFIN
1 → NDELTA
22400. → TOTSPL
15A
14C
11200 → NFIN
224 → NDELTA
100. → TOTSPL
15A
14D
33 → NFIN
1 → NDELTA
33. → TOTSPL
1 → KX
ANUM(KX) → BUFFER(KX)
IS KX = 224
NO
KX+1 → KX
YES
1 → KX
ANUM(KX+224) → ANUM(KX)
IS KX = 10976
NO
KX+1 → KX
YES
14E

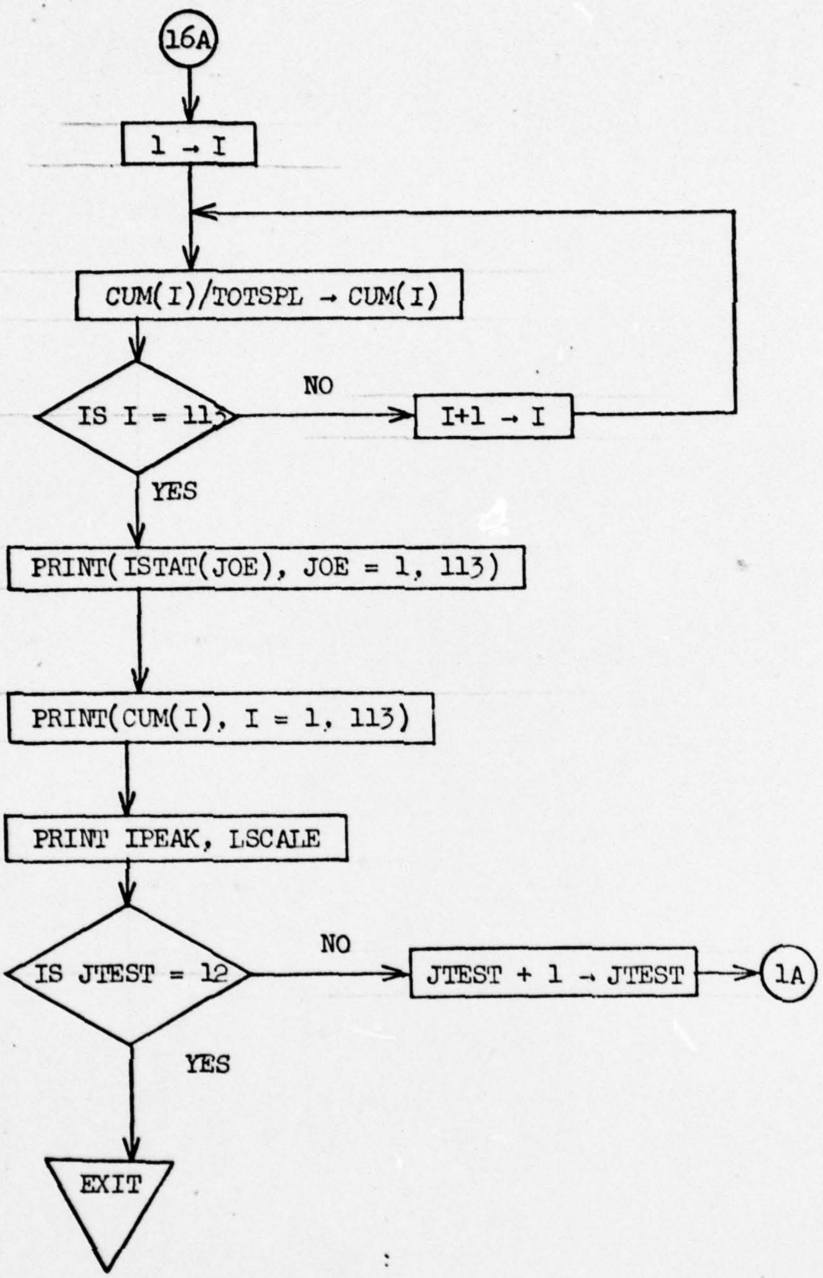

14E
1 → I
672 X I → IXX
IXX - 670 → IX
ANUM(IX-1) → ANUM(I)
IX → J
AMAX1(ANUM(I), ANUM(J)), → ANUM(I)
IS J = IXX
NO
J+1 → J
YES
IS I = 16
NO
I+1 → I
YES
1 → KX
ANUM(KX) → TEMP(KX)
IS KX = 16
NO
KX+1 → KX
YES
REWIND 3
14F


14F
1 → IK
READ(3)(IBUFER(L), L = 1, 224)
IS IK = NFILE
NO
IK+1 → IK
READ(3)(ANUM(L), L = 1, 11200)
1 → KXX
11201 → KX
BUFFER(KXX) → ANUM(KX)
KXX + 1 → KXX
IS KX = 11424
NO
KX+1 → KX
YES
1 → I
14H
672 X J → IXX
IXX - 670 → IX
ANUM(IX-1) → ANUM(I)
IX → J
14G

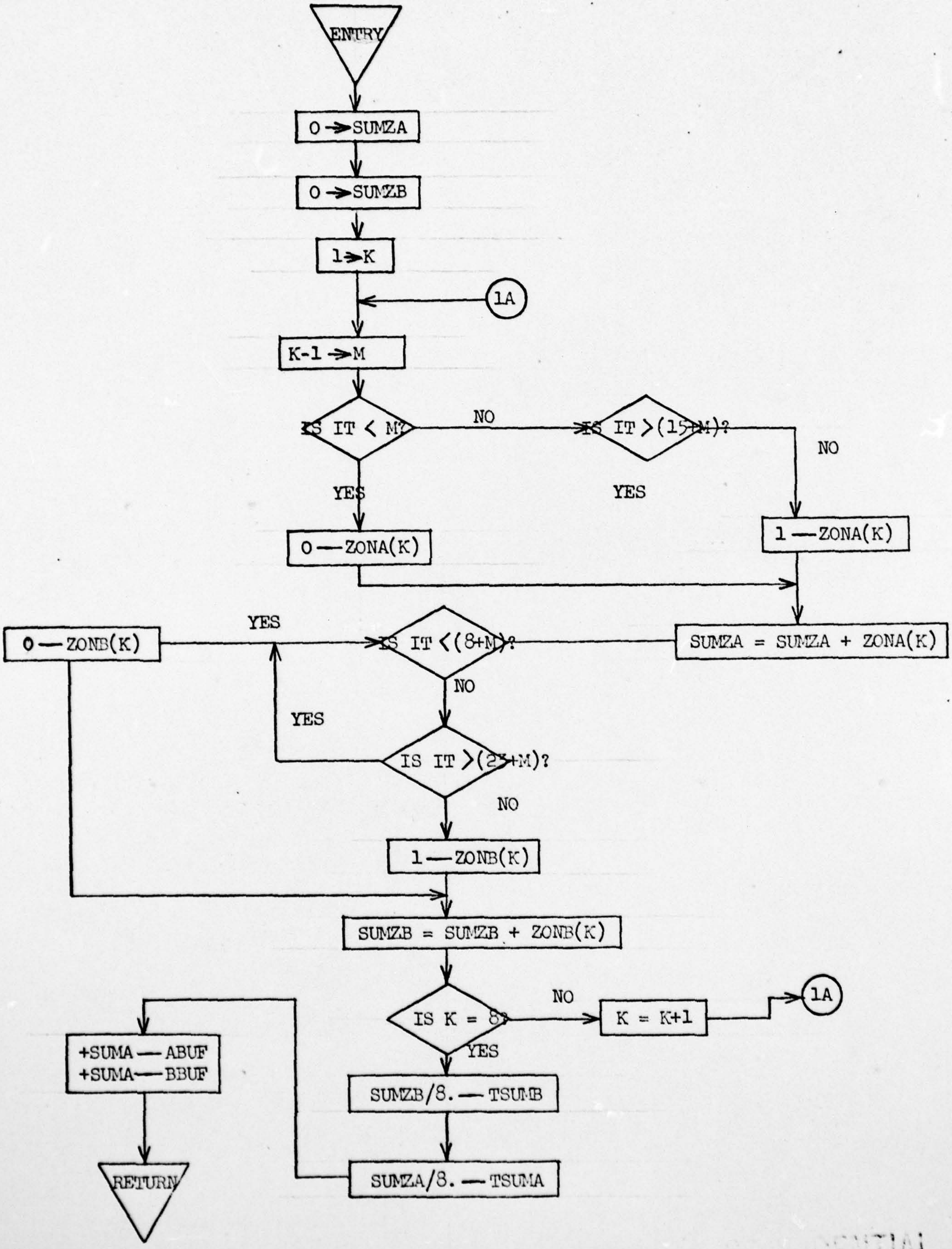

14G
AMAX1(ANUM(I), ANUM(J)) → ANUM(I)
IS J = IEX
NO
J+1 → J
YES
IS I = 17
NO
14H
YES
18 → KX
TEMP(KXX-17) → ANUM(KX)
IS KX = 33
NO
KX + 1 → KX
YES
1 → JK
ANUM(JK) - 111. → INDEX
ISTAT(INDEX)+ 1 → ISTAT(INDEX)
IS JK = 33
NO
JK+1 → JK
15C


15A
IS LSCALE > 1
YES
11424 → NFIN
NO
1 → JK
ANUM(JK) - 111. → INDEX
ISTAT(INDEX) + 1 → ISTAT(INDEX)
IS JK ≥ NFIN
NO
JK + NDELTA → JK
YES
REWIND3
1 → IK
READ(3)(BUFFER(KO), KO = 1, 224)
IS IK = NFILE
NO
IK+1 → IK
YES
READ(3)(ANUM(L), L = 1, 11200)
15B


15B
1 → JK
ANUM(JK) - 111. → INDEX
ISTAT(INDEX)+ 1 → ISTAT(INDEX)
IS JK > 11200
NO
JK + NDELTA → JK
YES
15C
ISTAT(113) → CUM(113)
Ø → JAA
1 → I
113 - I → JA
ISTAT(JA) → STAT
JA+1 → JAA
CUM(JAA) + STAT → CUM(JA)
IS I = 112
NO
I+1 → I
YES
16A


16A
1 → I
CUM(I)/TOTSPL → CUM(I)
IS I = 113
NO
I+1 → I
YES
PRINT(ISTAT(JOE), JOE = 1, 113)
PRINT(CUM(I), I = 1, 113)
PRINT IPEAK, LSCALE
IS JTEST = 12
NO
JTEST + 1 → JTEST
1A
YES
EXIT

SUBROUTINE ZONEIT (IT, ABUF, BBUF)



APPENDIX C

Program Listings

PROGRAM LISTING:

Noise Subroutine
Single Precision

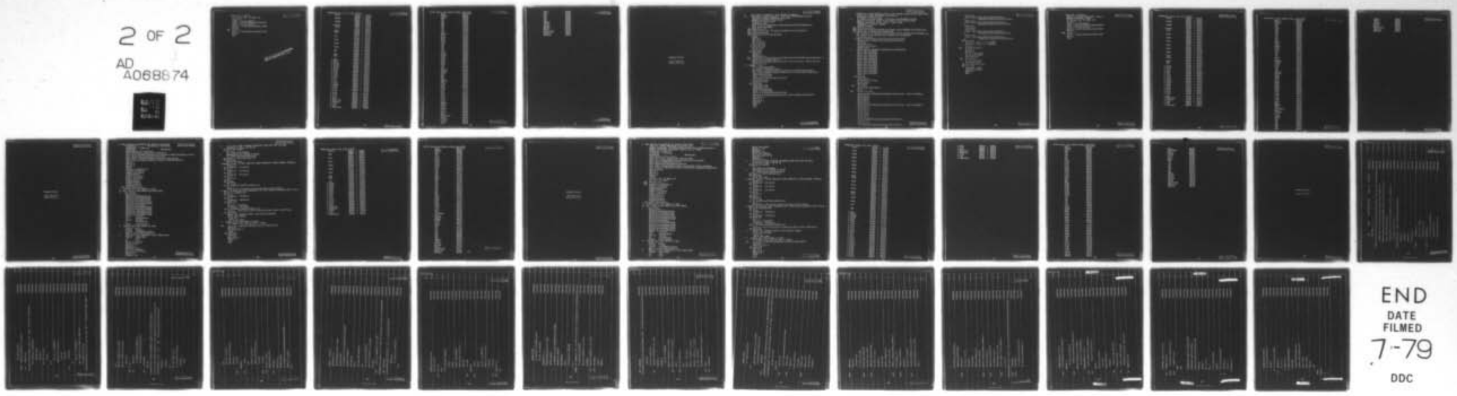
```
C      PAIR SYSTEM ANALYSIS  WAVE PERIØD PRØCESSØR
 C      FILTERED GAUSSIAN NØISE - FIVE PØLE BUTTERWØRTH FILTER
       DØUBLE PRECISIØN XS(5),XC(5)
       DIMENSIØN YS(3),YC(3)
       INTEGER RNDSET
       DATA NTRY/0/,RNDSET/Ø1/,LNGTH/288/,MULTP/56/,NBLKS/101/,BLKS/101./
   600 FØRMAT(1H1//)
   601 FØRMAT(5(1X,I9), 8X,Ø12/2(5D23.16/),2(3E23.8/))
   602 FØRMAT(4G24.8/)
   603 FØRMAT(5X,I5,8X,Ø12,8X,Ø12)
       PAUSE
       REWIND 1
       REWIND 2
       DØ  5 I=1,5
       XS(I)=0.0D+0
     5 XC(I)=0.0D+0
       DØ 10 I=1,3
       YS(I)=0.0
    10 YC(I)=0.0
       SUMZ=0.0
       SSQZ=0.0
       WRITE(6,600)
 C   NBLØCK = 101
       DØ 15 NBLØCK=1,NBLKS
       J = MØD(2*(NBLØCK-1),5)+0
       WRITE(6,601)NBLØCK,LNGTH,MULTP,J,NTRY,RNDSET,XS,XC,YS,YC
       PAUSE
       CALL BLØCK(NBLØCK,LNGTH,J,NTRY,RNDSET,XS,XC,YS,YC,YBAR,YMSQ)
       VARY=YMSQ-YBAR**2
       SIGY=SQRT(VARY)
       WRITE(6,602)YBAR,YMSQ,VARY,SIGY
       SUMZ=SUMZ+YBAR
       SSQZ=SSQZ+YMSQ
    15 CØNTINUE
       ZBAR=SUMZ/BLKS
       ZMSQ=SSQZ/BLKS
       VARZ=ZMSQ-ZBAR**2
       SIGZ=SQRT(VARZ)
       WRITE(6,602)ZBAR,ZMSQ,VARZ,SIGZ
       WRITE(5,601)NBLØCK,LNGTH,MULTP,J,NTRY,RNDSET,XS,XC,YS,YC
       END FILE 1
       REWIND 1
       END FILE 2
       REWIND2
       STØP
       END
```

```
      SUBRØUTINE BLØCK(NBLØCK,LNGTH,J,NTRY,RNDSET,XS,XC,YS,YC,YBAR,YMSQ)
      DIMENSIØN L1(56),L2(56)
C     FILTERED GAUSSIAN NØISE - FIVE PØLE BUTTERWØRTH FILTER
      DØUBLE PRECISIØN XS(5),XC(5),TEMP,T,BW,C1,C2,C3,C4,C5
      DIMENSIØN YS(3),YC(3),Y(16132),G(288)
      INTEGER RNDSET
  601 FØRMAT(5(2XI18))
  503 FØRMAT(2(4XE23.16)/3XE23.16,2(4XE23.16)/2(4XE23.16)/4(4XE15.8))
  603 FØRMAT(4H T =E23.16,4H BW=E23.16/
     14H C1=E23.15,4H C2=E23.16,4H C3=E23.16/4H C4=E23.16,4H C5=E23.16/
     24H C6=E14.8,4H C7=E14.8,4H C8=E14.8,4H C9=E14.8/)
      IF(NTRY)5,5,15
    5 READ (5,503)T,BW,C1,C2,C3,C4,C5,C6,C7,C8,C9
      WRITE(6,603)T,BW,C1,C2,C3,C4,C5,C6,C7,C8,C9
      MULTP=56
      MLTM1=MULTP-1
      NUMBR=MULTP*LNGTH
      DENØM=NUMBR
C     THREE PØINT LAGRANGE INTERPØLATIØN CØNSTANTS
      SA1=-.08*.95105652
      SA2=+.96*.95105652
      SA3=+.12*.95105652
      CA1=-.08*.30901699
      CA2=+.96*.30901699
      CA3=+.12*.30901699
      SB1=-.12*.58778525
      SB2=+.84*.58778525
      SB3=+.28*.58778525
      CB1=-.12*.80901699
      CB2=+.84*.80901699
      CB3=+.28*.80901699
      NTRY =1
   15 IFLAG=0
      IMAX =3225
      IF(J.EQ.0)GØ TØ 25
      DØ 20 I=1,J
      M=I+NUMBR
   20 Y(I)=Y(M)
      IF(J.GT.2)IMAX=IMAX-1
      II=0
   25 DØ 30 I=0,IMAX
      TEMP=C1*XS(5)-C2*XS(4)+C3*XS(3)-C4*XS(2)+C5  *XS(1)+GAUSS(2)
      XS(1)=XS(2)
      XS(2)=XS(3)
      XS(3)=XS(4)
      XS(4)=XS(5)
      XS(5)=TEMP
      TEMP=C1*XC(5)-C2*XC(4)+C3*XC(3)-C4*XC(2)+C5  *XC(1)+GAUSS(1)
      XC(1)=XC(2)
      XC(2)=XC(3)
      XC(3)=XC(4)
      XC(4)=XC(5)
      XC(5)=TEMP
      YS(1)=YS(2)
      YS(2)=YS(3)
      YS(3)=C6*SNGL(XS(4))+C7*SNGL(XS(3))+C8*SNGL(XS(2))+C9*SNGL(XS(1))
      YC(1)=YC(2)
      YC(2)=YC(3)
      YC(3)=C6*SNGL(XC(4))+C7*SNGL(XC(3))+C8*SNGL(XC(2))+C9*SNGL(XC(1))
      M=5*I+J+1
      Y(M)=-(SB3*YS(1)+SB2*YS(2)+SB1*YS(3)
     1             +CB3*YC(1)+CB2*YC(2)+CB1*YC(3))
```

```
      M=5*I+J+2
      Y(M)=-(SA3*YS(1)+SA2*YS(2)+SA1*YS(3))
     2              +CA3*YC(1)+CA2*YC(2)+CA1*YC(3)
      M=5*I+J+3
      Y(M)=YC(2)
      M=5*I+J+4
      Y(M)=  SA1*YS(1)+SA2*YS(2)+SA3*YS(3)
     4              +CA1*YC(1)+CA2*YC(2)+CA3*YC(3)
      M=5*I+J+5
      Y(M)=  SB1*YS(1)+SB2*YS(2)+SB3*YS(3)
     5              -(CB1*YC(1)+CB2*YC(2)+CB3*YC(3))
   30 CØNTINUE
      WRITE(1)  (Y(M), M = 1,NUMBR)
      WRITE(2)  (Y(M), M = 1,NUMBR)
      II=II+1
701   RNDSET=IRAND3(1)
      SUMY=0.0
      SSQY=0.0
      DØ 85 K =0,MLTM1
      DØ 32 K1=1,LNGTH
      M=K*LNGTH+K1
 32   G(K1)=Y(M)
      DØ 45 I=1,LNGTH
      SUMY=SUMY+G(I)
   45 SSQY=SSQY+G(I)**2
   85 CØNTINUE
      YBAR=SUMY/DENØM
      YMSQ=SSQY/DENØM
      RETURN
      END
```

AD-A068 874   NAVAL UNDERSEA WARFARE CENTER SAN DIEGO CA   F/G 17/1
ADAPTATION OF COMPUTER PROGRAMS FOR THE DIGITAL SIMULATION OF T--ETC(U)
JUL 68 J MUNOZ-FLORES, S K BUEHLER

UNCLASSIFIED   NUWC-TN-149   NL



2 OF 2

AD
A068874

END
DATE
FILMED
7-79
DDC

```
      FUNCTION GAUSS(AA)
       IF (IFLAG .NE. 0) GOTO 100
       IFLAG = 1
       TRANR = RANDM1(DUMMY)
       TEMP = SQRT(2.*ABS(ALOG(TRANR)))
       ANGLE = RANDM1(DUMMY)
       GAUSS = SIN(6.283185*ANGLE)*TEMP
       RETURN
  100  IFLAG = 0
       GAUSS = COS(6.283185*ANGLE)*TEMP
       RETURN
       END
```

```
PRØGRAMS LØADED AND CØRE LIMITS
        MAIN1                    00600   -   01201
                                 01202   -   01276
        BLØCK                    01277   -   03211
                                 03212   -   43561
        GAUSS                    43562   -   43651
                                 43652   -   43665
      * RANDM                    43666   -   43726
        AI62                     43727   -   43743
                                 43744   -   43745
        DBLE                     43746   -   43757
        CØS                      43760   -   43774
                                 43775   -   43776
        SIN                      43777   -   44120
                                 44121   -   44144
        EI21                     44145   -   44163
                                 44164   -   44165
        EI22                     44166   -   44276
                                 44277   -   44311
        ALØG                     44312   -   44407
                                 44410   -   44423
        EXP                      44424   -   44534
                                 44535   -   44557
        ABS                      44560   -   44573
        MØD                      44574   -   44620
                                 44621   -   44621
      * SQRT                     44622   -   44736
      * SNGL                     44737   -   45005
      * MCI61                    45006   -   45026
      * MCI62                    45027   -   45055
      * ASI22                    45056   -   45205
      * MI22                     45206   -   45277
      * DI22                     45300   -   45376
      * NI22                     45377   -   45403
      * CI21                     45404   -   45431
      * CI71                     45432   -   45440
      * AINT                     45441   -   45467
      * IAT                      45470   -   45525
      * AI71                     45526   -   45543
      * CI26                     45544   -   45605
      * ASI66                    45606   -   46034
      * MI66                     46035   -   46232
      * CI62                     46233   -   46266
      * CI61                     46267   -   46322
      * CI12                     46323   -   46367
      * ZI70                     46370   -   46405
      * CIVT                     46406   -   46461
      * ARL1                     46462   -   46517
      * EXIT/1ST                 46520   -   46521
      * FØRTI/Ø                  46522   -   52242
      * FIER                     52243   -   52327
      * IPA                      52330   -   52345
      * FØRT/MS-1                52346   -   53003
```

CONFIDENTIAL

```
ENTRY PØINT AND CØMMØN BLØCK LØCATIØNS
                NAME              LØCATIØN
                FBLANK             00000
                MAINI              00602
                IPA                52330
                IRE                52162
                IWR                46612
                IND                47340
                MØD                44576
                IIX                47276
                IID                47265
                IIF                47272
                BLØCK              01301
                EI21               44147
                SI22               45056
                SQRT               44622
                AI22               45064
                DI22               45300
                IEN                52155
                IST                46520
                IAT                45470
                IRD                46605
                CI12               46323
                MI22               45206
                GAUSS              43564
                MI66               46035
                SI66               45606
                AI66               45627
                AI62               43731
                SNGL               44737
                I21                51251
                INE                51330
                IRAND3             43722
                RANDM1             43666
                ALØG               44314
                ABS                44562
                SIN                44001
                CØS                43762
                RANDM2             43712
                DBLE               43750
                CI26               45544
                CI21               45404
                NI22               45377
                EI22               44170
                EXP                44426
                AINT               45441
                FIER               52243
                CI71               45432
                ZI70               46370
                AI71               45526
                ERRØR              52255
                IDINT              44754
                CI62               46233
                MCI62              45027
                CI61               46267
                MCI61              45006
                ARL1               46462
                CIVTARGL1          46444
                ARGL1              46505
                CIVTAQ             46424
                CIVTQ              46406
                IIFA               45502
```

CONFIDENTIAL

CONFIDENTIAL

```
MØNII              74506
IBA                52150
IIE                47260
IDE                46527
IEC                46522
IEF                46566
IØRETI             46765
IØTBLI             52636
WRITEI             52364
SETEØF             52346
READI              52361
BACKSPACEI         52353
ENDFILEI           52356
REWINDI            52367
```

CONFIDENTIAL

PROGRAM LISTING:

Noise Subroutine
Double Precision

```
C     PAIR SYSTEM ANALYSIS  WAVE PERIØD PRØCESSØR
C      FILTERED GAUSSIAN NØISE - FIVE PØLE BUTTERWØRTH FILTER
       IMPLICIT DØUBLE PRECISIØN (B,S,U-Z)
       DØUBLE PRECISIØN XS(5),XC(5)
       DIMENSIØN YS(3),YC(3)
       INTEGER RNDSET
       DATA NTRY/0/,RNDSET/Ø1/,LNGTH/288/,MULTP/56/,NBLKS/101/
       DATA BLKS/101.0D00/
   600 FØRMAT(1H1//)
   601 FØRMAT(5(1X,I9), 8X,Ø12/2(1X,5D23.16/),2(3E23.8/))
   602 FØRMAT(4D24.8/)
   603 FØRMAT(5X,I5,8X,Ø12,8X,Ø12)
       PAUSE
       REWIND 1
       REWIND 2
       DØ  5 I=1,5
       XS(I)=0.0D+0
     5 XC(I)=0.0D+0
       DØ 10 I=1,3
       YS(I)=0.0
    10 YC(I)=0.0
       SUMZ=0.0
       SSQZ=0.0
       WRITE(6,600)
  503  FØRMAT(8X,Ø12/3(3X,D23.16)/2(3X,D23.16)/3(3X,D23.16)/2(3X,D23.16)
      1/3(3X,D23.8)/3(3X,D23.8)/)
       READ(5,503)RNDSET,(XS(I),I=1,5), (XC(I),I=1,5), (YS(I),I=1,3),
      1(YC(I),I=1,3)
C   NBLØCK = 101
       DØ 15 NBLØCK=1,NBLKS
       J = MØD(2*(NBLØCK-1),5)+0
       WRITE(6,601)NBLØCK,LNGTH,MULTP,J,NTRY,RNDSET,XS,XC,YS,YC
       CALL BLØCK(NBLØCK,LNGTH,J,NTRY,RNDSET,XS,XC,YS,YC,YBAR,YMSQ)
       VARY=YMSQ-YBAR**2
       SIGY = DSQRT(VARY)
       WRITE(6,602)YBAR,YMSQ,VARY,SIGY
       SUMZ=SUMZ+YBAR
       SSQZ=SSQZ+YMSQ
    15 CØNTINUE
       ZBAR=SUMZ/BLKS
       ZMSQ=SSQZ/BLKS
       VARZ=ZMSQ-ZBAR**2
       SIGZ = DSQRT(VARZ)
       WRITE(6,602)ZBAR,ZMSQ,VARZ,SIGZ
       WRITE(6,601)NBLØCK,LNGTH,MULTP,J,NTRY,RNDSET,XS,XC,YS,YC
       END FILE 1
       REWIND 1
       END FILE 2
       REWIND2
       STØP
       END
      SUBRØUTINE BLØCK(NBLØCK,LNGTH,J,NTRY,RNDSET,XS,XC,YS,YC,YBAR,YMSQ)
       IMPLICIT DØUBLE PRECISIØN (A-H,Ø-X,Z)
       DIMENSIØN L1(56),L2(56)
C      FILTERED GAUSSIAN NØISE - FIVE PØLE BUTTERWØRTH FILTER
       DØUBLE PRECISIØN XS(5),XC(5),TEMP,T,BW,C1,C2,C3,C4,C5
       DØUBLE PRECISIØN YS(3), YC(3), YBAR, YMSQ
       DIMENSIØN Y(16132), G(288)
       INTEGER RNDSET
   601 FØRMAT(5(2XI18))
   503 FØRMAT(2(4XE23.16)/3XE23.16,2(4XE23.16)/2(4XE23.16)/4(4XE15.8))
   603 FØRMAT(4H T =E23.16,4H BW=E23.16/
      14H C1=E23.16,4H C2=E23.16,4H C3=E23.16/4H C4=E23.16,4H C5=E23.16/
      24H C6=E15.8,4H C7=E15.8,4H C8=E15.8,4H C9=E15.8/)
       IF(NTRY)5,5,15
     5 READ (5,503)T,BW,C1,C2,C3,C4,C5,C6,C7,C8,C9
       WRITE(6,603)T,BW,C1,C2,C3,C4,C5,C6,C7,C8,C9
       MULTP=56
       MLTM1=MULTP-1
       NUMBR=MULTP*LNGTH
       DENØM=NUMBR
C      THREE PØINT LAGRANGE INTERPØLATIØN CØNSTANTS
       SA1=-.08*.95105652
       SA2=+.96*.95105652
       SA3=+.12*.95105652
       CA1=-.08*.30901699
       CA2=+.96*.30901699
       CA3=+.12*.30901699
       SB1=-.12*.58778525
       SB2=+.84*.58778525
       SB3=+.28*.58778525
       CB1=-.12*.80901699
       CB2=+.84*.80901699
       CB3=+.28*.80901699
       CALL RANDM2(RNDSET)
       NTRY =1
    15 IFLAG=0
       IMAX =3225
       IF(J.EQ.0)GØ TØ 25
       DØ 20 I=1,J
       M=I+NUMBR
   20  Y(I)=Y(M)
       IF(J.GT.2)IMAX=IMAX-1
       I1=0
   25  DØ 30 I=0,IMAX
       TEMP=C1*XS(5)-C2*XS(4)+C3*XS(3)-C4*XS(2)+C5  *XS(1)+GAUSS(2)
       XS(1)=XS(2)
       XS(2)=XS(3)
       XS(3)=XS(4)
       XS(4)=XS(5)
       XS(5)=TEMP
       TEMP=C1*XC(5)-C2*XC(4)+C3*XC(3)-C4*XC(2)+C5  *XC(1)+GAUSS(1)
       XC(1)=XC(2)
       XC(2)=XC(3)
       XC(3)=XC(4)
       XC(4)=XC(5)
       XC(5)=TEMP
       YS(1)=YS(2)
       YS(2)=YS(3)
       YS(3)=C6*XS(4)+C7*XS(3)+C8*XS(2)+C9*XS(1)
       YC(1)=YC(2)
       YC(2)=YC(3)
       YC(3)=C6*XC(4)+C7*XC(3)+C8*XC(2)+C9*XC(1)
      M=5*I+J+1
      Y(M)=-(SB3*YS(1)+SB2*YS(2)+SB1*YS(3)
     1               +CB3*YC(1)+CB2*YC(2)+CB1*YC(3))
      M=5*I+J+2
      Y(M)=-(SA3*YS(1)+SA2*YS(2)+SA1*YS(3))
     2               +CA3*YC(1)+CA2*YC(2)+CA1*YC(3)
      M=5*I+J+3
      Y(M)=YC(2)
      M=5*I+J+4
      Y(M)=  SA1*YS(1)+SA2*YS(2)+SA3*YS(3)
     4               +CA1*YC(1)+CA2*YC(2)+CA3*YC(3)
      M=5*I+J+5
      Y(M)=  SB1*YS(1)+SB2*YS(2)+SB3*YS(3)
     5               -(CB1*YC(1)+CB2*YC(2)+CB3*YC(3))
   30 CONTINUE
      WRITE(1) (Y(M), M = 1,NUMBR)
      WRITE(2) (Y(M), M = 1,NUMBR)
      II=II+1
701   RNDSET=IRAND3(1)
      SUMY=0.0
      SSQY=0.0
      DO 85 K =0,MLTM1
      DO 32 K1=1,LNGTH
      M=K*LNGTH+K1
 32   G(K1)=Y(M)
      DO 45 I=1,LNGTH
      SUMY=SUMY+G(I)
   45 SSQY=SSQY+G(I)**2
   85 CONTINUE
      YBAR=SUMY/DENOM
      YMSQ=SSQ./DENOM
      RETURN
      END
      FUNCTIØN GAUSS(AA)
       IMPLICIT DØUBLE PRECISIØN (A,G,T)
       DØUBLE PRECISIØN GAUSS
       IF (IFLAG .NE. 0) GØTØ 100
       IFLAG = 1
       TRANR = RANDM1(DUMMY)
       TEMP = DSQRT(2.*DABS(DLØG(TRANR)))
       ANGLE = RANDM1(DUMMY)
       GAUSS = DSIN(6.283185*ANGLE)*TEMP
       RETURN
 100   IFLAG = 0
       GAUSS = DCØS(6.283185*ANGLE)*TEMP
       RETURN
       END
```

CONFIDENTIAL

```
PRØGRAMS LØADED AND CØRE LIMITS
          MAINI          00600   -   01347
                         01350   -   01471
          BLØCK          01472   -   03576
                         03577   -   44627
          GAUSS          44630   -   44735
                         44736   -   44757
        * RANDM          44760   -   45020
          EI61           45021   -   45042
                         45043   -   45046
          EI66           45047   -   45214
                         45215   -   45234
          DCØS           45235   -   45253
                         45254   -   45257
          DSIN           45260   -   45424
                         45425   -   45475
          DLØG           45476   -   45612
                         45613   -   45661
          DEXP           45662   -   46031
                         46032   -   46075
          DMØD           46076   -   46125
                         46126   -   46127
          DABS           46130   -   46144
          MI62           46145   -   46161
                         46162   -   46163
          DBLE           46164   -   46175
          MØD            46176   -   46222
                         46223   -   46223
        * DSQRT          46224   -   46327
        * MI22           46330   -   46421
        * CI81           46422   -   46431
        * DINT           46432   -   46467
        * IAT            46470   -   46525
        * CI26           46526   -   46567
        * ZI80           46570   -   46614
        * AI81           46615   -   46645
        * ASI66          46646   -   47074
        * MI66           47075   -   47272
        * DI66           47273   -   47500
        * NI66           47501   -   47511
        * CI62           47512   -   47545
        * CI61           47546   -   47601
        * CI16           47602   -   47627
        * CIVT           47630   -   47703
        * ARL1           47704   -   47741
        * EXIT/IST       47742   -   47743
        * FØRTI/Ø        47744   -   53464
        * FIER           53465   -   53551
        * IPA            53552   -   53567
        * FØRT/MS-1      53570   -   54225
```

CONFIDENTIAL

CONFIDENTIAL

```
ENTRY PØINT AND CØMMØN BLØCK LØCATIØNS
                NAME                    LØCATIØN
                FBLANK                  00000
                MAINI                   00602
                IPA                     53552
                IRE                     53404
                C126                    46526
                IWR                     50034
                IND                     50562
                IRD                     50027
                IIX                     50520
                IID                     50507
                MØD                     46200
                BLØCK                   01474
                E161                    45023
                S166                    46646
                DSQRT                   46224
                A166                    46667
                D166                    47273
                IEN                     53377
                IST                     47742
                IAT                     46470
                C116                    47602
                M122                    46330
                RANDM2                  45004
                GAUSS                   44632
                M166                    47075
                C162                    47512
                I2I                     52473
                INE                     52552
                IRAND3                  45014
                RANDMI                  44760
                DLØG                    45500
                DABS                    46132
                M162                    46147
                DSIN                    45262
                DCØS                    45237
                E166                    45051
                DEXP                    45664
                DMØD                    46100
                FIER                    53465
                C161                    47546
                N166                    47501
                C181                    46422
                Z180                    46570
                A181                    46615
                DINT                    46432
                DBLE                    46166
                ERRØR                   53477
                ARLI                    47704
                CIVTARGLI               47666
                ARGLI                   47727
                CIVTAQ                  47646
                IIFA                    46502
                CIVTQ                   47630
                MØNII                   74506
                IBA                     53372
                IIF                     50514
                IIE                     50502
                IDE                     47751
                IEC                     47744
                IEF                     50010
```

CONFIDENTIAL

```
IØRETI                 50207
IØTBLI                 54060
WRITEI                 53606
SETEØF                 53570
READI                  53603
BACKSPACEI             53575
ENDFILEI               53600
REWINDI                53611
```

PROGRAM LISTING:

Wave Subroutine
Single Precision

```
C  WAVE PERIØD MEASURMENTS ØF SIGNAL PLUS NØISE
      DIMENSIØN           TT(224,12), SCALE(12)
      DIMENSIØN  IT(224,12)
      DIMENSIØN                             SPN(16128)
      EQUIVALENCE (IT(1),TT(1))
      DATA LØUT/224/, LSCALE/12/, FNP/12.0/
      DATA PI/3.1415927/,F0/20000.0/,FBW/440.0/,DELT/0.000010/,NP/12/
      DATA NBLKS/101/,LNGTH/288/,MULTP/56/
      DATA RMS/1.0004576/,SQRT2/1.4142136/,TP/0.16128/
      DATA ØNEDB/1.122018/,THRDB/1.412537/,TENDB/3.1622777/
      REWIND 1
      REWIND 2
      REWIND 3
      PAUSE
      DB1PHF=1.122018**1.5
      TWØDB=1.122018**2
      F3DB=1.122018**3
      F4DB=TWØDB**2
      F5DB=1.122018**5
      F6DB = THRDB**2
      F12DB=F6DB**2
      F18DB=F6DB**3
      F24DB=F12DB**2
      F30DB=TENDB**3
  611 FØRMAT(5(2XI18)/)
C    CHANGE THIS FØR CHANGE IN LØUT
C  S/N = PEAK SIGNAL/SQRT(2.0)/RMS NØISE
C     S/N RATIØS
      SCALE(1)=0.0
      SCALE(2)=RMS*SQRT2/F6DB
      SCALE(3)=RMS*SQRT2/F4DB
      SCALE(4)=RMS*SQRT2/F3DB
      SCALE(5)=RMS*SQRT2/TWØDB
      SCALE(6)=RMS*SQRT2/ØNEDB
      SCALE(7)=RMS*SQRT2
      SCALE(8)=RMS*SQRT2*ØNEDB
      SCALE(9)=RMS*SQRT2*TWØDB
      SCALE(10)=RMS*SQRT2*F3DB
      SCALE(11)=RMS*SQRT2*F4DB
      SCALE(12)=RMS*SQRT2*F6DB
      MLTM1     =MULTP-1
      LTT       =LSCALE*LØUT
      TWØ PI    =2.0*PI
      F1        =0.5*FBW/TP
      NUMBR     =MULTP*LNGTH
C    CHANGE THIS FØR CHANGE IN LØUT
      NUMBR = 16128
      FNUMBR    =NUMBR
      TNUMBR    =(FNUMBR+1.0)/2.0
      ØSCL = 32.0*F0*F0/(FNP*FBW)
C     DIGITAL CLØCK FREQUENCY 2.424 MEGACYCLES
      TSCALE    =DELT*ØSCL
      L2        =LTT
      L6        =LTT
      DØ 60 N=1,NBLKS
      READ(1) SPN
      REWIND 2
      WRITE(2) SPN
      DØ 55 L=1,LSCALE
      IF(L.EQ.1)GØ TØ 23
      REWIND 2
      READ(2) SPN
```

```
C       19.2 KYDS RANGE (CHANGE FØLLØWING CARD FØR DIFF RB BIN)
        IF(MØD(N.3).NE.1) GØ TØ 30
     23 DØ 25 J=1,NUMBR
        SJ = J
        TX = DELT*(SJ-TNUMBR)
C       GENERAATIØN ØF LINEAR FM PULSE
        S = CØS(TWØPI*TX*(FO+F1*TX))
     25 SPN(J) = SPN(J)+SCALE(L)*S
     30 CØNTINUE
        DØ 55 I=1,LØUT
C CØMPUTE START = FIRST PØSITIVE ZERØ CRØSSING WITHIN SAMPLE INTERVAL
        J=72*I-71
     40 IF(SPN(J)   )41.42.43
     41 J=J+1
        IF(SPN(J)   )41.45.46
     42 J=J+1
        IF(SPN(J)   )41.45.44
     43 J=J+1
        GØ TØ 40
     44 J=J-1
     45 START=J
        GØ TØ 50
     46 P LIN=SPN(J)/(SPN(J)-SPN(J-1))
        SJ = J
        START=(((P LIN-1.5)*P LIN+0.5)*0.566-1.0)*P LIN+SJ
C CØUNT NP PØSITIVE ZERØ CRØSSINGS AND THEN CØMPUTE ELAPSED TIME TT(I.L)
     50 DØ 53 ICRØSS=1.NP
     51 J=J+1
        IF(SPN(J)   )52.52.51
     52 J=J+1
        IF(SPN(J)   )52.53.53
     53 CØNTINUE
        SJ = J
        TT(I.L) = SJ-START
        IF(SPN(J)   )54.55.54
     54 P LIN=SPN(J)/(SPN(J)-SPN(J-1))
        TT(I.L)   =(((P LIN-1.5)*P LIN+0.5)*0.566-1.0)*P LIN+TT(I,L)
     55 CØNTINUE
C       MØDULØ 32 CØUNTER ØUTPUT WITH SLIDING PRESET
        DØ 64 JK=1,LSCALE
         ISET =0
         DØ 64 I=1,LØUT
C      CHANGE THIS FØR CHANGE IN LØUT
        IF (MØD(I,7)+0 .EQ. 0) ISET = ISET+1
64      IT(I,JK) = MØD(INT(TSCALE*TT(I.JK))+ISET,32)+0
        WRITE(3) IT
        WRITE(6,511) N
        PAUSE
     60 CØNTINUE
        REWIND 1
        END FILE 3
        REWIND 3
        STØP
        END
```

```
PRØGRAMS LØADED AND CØRE LIMITS
       MAIN1                 00600  -  01604
                             01605  -  46556
       CØS                   46557  -  46573
                             46574  -  46575
       SIN                   46576  -  46717
                             46720  -  46743
       E121                  46744  -  46762
                             46763  -  46764
       E122                  46765  -  47075
                             47076  -  47110
       ALØG                  47111  -  47206
                             47207  -  47222
       EXP                   47223  -  47333
                             47334  -  47356
       ABS                   47357  -  47372
       MØD                   47373  -  47417
                             47420  -  47420
       INT                   47421  -  47432
     * AS122                 47433  -  47562
     * M122                  47563  -  47654
     * D122                  47655  -  47753
     * N122                  47754  -  47760
     * C121                  47761  -  50006
     * C171                  50007  -  50015
     * AINT                  50016  -  50044
     * IAT                   50045  -  50102
     * A171                  50103  -  50120
     * C112                  50121  -  50165
     * Z170                  50166  -  50203
     * CIVT                  50204  -  50257
     * ARL1                  50260  -  50315
     * EXIT/1ST              50316  -  50317
     * FØRTI/Ø               50320  -  54040
     * FIER                  54041  -  54125
     * IPA                   54126  -  54143
     * FØRT/MS-1             54144  -  54601
```

```
ENTRY PØINT AND CØMMØN BLØCK LØCATIØNS
                NAME                LØCATIØN
                FBLANK               00000
                MAINI                00602
                IRE                  53760
                IPA                  54126
                EI22                 46767
                EI21                 46746
                MI22                 47563
                DI22                 47655
                CI12                 50121
                AI22                 47441
                IRD                  50403
                I2I                  53047
                INE                  53126
                IWR                  50410
                MØD                  47375
                SI22                 47433
                CØS                  46561
                INT                  47423
                IIX                  51074
                IND                  51136
                IEN                  53753
                IST                  50316
                IAT                  50045
                SIN                  46600
                ABS                  47361
                CI21                 47761
                NI22                 47754
                ALØG                 47113
                EXP                  47225
                AINT                 50016
                FIER                 54041
                CI71                 50007
                ZI70                 50166
                AI71                 50103
                ARLI                 50260
                CIVTARGLI            50242
                ARGLI                50303
                CIVTAQ               50222
                ERRØR                54053
                CIVTQ                50204
                IIFA                 50057
                MØNII                74506
                IBA                  53746
                IIF                  51070
                IID                  51063
                IIE                  51056
                IDE                  50325
                IEC                  50320
                IEF                  50364
                IØRETI               50563
                IØTBLI               54434
                WRITEI               54162
                SETEØF               54144
                READI                54157
                BACKSPACEI           54151
                ENDFILEI             54154
                REWINDI              54165
```

PROGRAM LISTING:

Wave Subroutine
Double Precision

```
C  WAVE PERIØD MEASURMENTS ØF SIGNAL PLUS NØISE
      IMPLICIT DØUBLE PRECISIØN (A-H,Ø-R,U-Z)
      DØUBLE PRECISIØN TP,THRDB,TENDB,TWØDB,TWØPI,TNUMBR,TSCALE,TX
      DØUBLE PRECISIØN SCALE(12), SQRT2, SJ, S, START
      DIMENSIØN TT(224,12)
      DIMENSIØN  IT(224,12)
      DIMENSIØN                         SPN(16128)
      EQUIVALENCE (IT(1),TT(1))
      DATA LØUT/224/, LSCALE/12/, FNP/12.0D00/
      DATA PI/3.1415927D00/,FO/20000.0D00/,FBW/440.0D00/
      DATA DELT/0.000010D00/,NP/12/
      DATA NBLKS/5/,LNGTH/288/,MULTP/56/
      DATA RMS/1.0044810D00/,SQRT2/1.4142136D00/,TP/0.16128D00/
      DATA ØNEDB/1.122018D00/,THRDB/1.412537D00/,TENDB/3.1622777D00/
      DATA NSKP/95/
      REWIND 1
      REWIND 2
      REWIND 3
      PAUSE
      IF(NSKP .EQ. 0) GØTØ 200
      DØ 100 I = 1,NSKP
  100 READ(1) SPN
  200 DB1PHF=1.122018**1.5
      TWØDB=1.122018**2
      F3DB=1.122018**3
      F4DB=TWØDB**2
      F5DB=1.122018**5
      F6DB = THRDB**2
      F12DB=F6DB**2
      F18DB=F6DB**3
      F24DB=F12DB**2
      F30DB=TENDB**3
  611 FØRMAT(5(2XI18)/)
C   CHANGE THIS FØR CHANGE IN LØUT
C  S/N = PEAK SIGNAL/SQRT(2.0)/RMS NØISE
C     S/N RATIØS
      SCALE(1)=0.0
      SCALE(2)=RMS*SQRT2/F6DB
      SCALE(3)=RMS*SQRT2/F4DB
      SCALE(4)=RMS*SQRT2/F3DB
      SCALE(5)=RMS*SQRT2/TWØDB
      SCALE(6)=RMS*SQRT2/ØNEDB
      SCALE(7)=RMS*SQRT2
      SCALE(8)=RMS*SQRT2*ØNEDB
      SCALE(9)=RMS*SQRT2*TWØDB
      SCALE(10)=RMS*SQRT2*F3DB
      SCALE(11)=RMS*SQRT2*F4DB
      SCALE(12)=RMS*SQRT2*F6DB
      MLTM1     =MULTP-1
      LTT       =LSCALE*LØUT
      TWØ PI    =2.0*PI
      F1        =0.5*FBW/TP
      NUMBR     =MULTP*LNGTH
C   CHANGE THIS FØR CHANGE IN LØUT
      NUMBR = 16128
      FNUMBR    =NUMBR
      TNUMBR    =(FNUMBR+1.0)/2.0
      ØSCL = 32.0*FO*FO/(FNP*FBW)
C     DIGITAL CLØCK FREQUENCY 2.424 MEGACYCLES
      TSCALE    =DELT*ØSCL
      L2        =LTT
      L6        =LTT
      DØ 60 N=1,NBLKS
      READ(1) SPN
      REWIND 2
      WRITE(2) SPN
      DØ 55 L=1,LSCALE
      IF(L.EQ.1)GØ TØ 23
      REWIND 2
      READ(2) SPN
C     19.2 KYDS RANGE (CHANGE FØLLØWING CARD FØR DIFF RB BIN)
      IF(MØD(N,3).NE.1) GØ TØ 30
   23 DØ 25 J=1,NUMBR
      SJ = J
      TX = DELT*(SJ-TNUMBR)
C     GENERAATIØN ØF LINEAR FM PULSE
      S = DCØS(TWØPI*TX*(FO+F1*TX))
   25 SPN(J) = SPN(J)+SCALE(L)*S
   30 CØNTINUE
      DØ 55 I=1,LØUT
C CØMPUTE START = FIRST PØSITIVE ZERØ CRØSSING WITHIN SAMPLE INTERVAL
      J=72*I-71
   40 IF(SPN(J)  )41,42,43
   41 J=J+1
      IF(SPN(J)  )41,45,46
   42 J=J+1
      IF(SPN(J)  )41,45,44
   43 J=J+1
      GØ TØ 40
   44 J=J-1
   45 START=J
      GØ TØ 50
   46 P LIN=SPN(J)/(SPN(J)-SPN(J-1))
      SJ = J
      START=(((P LIN-1.5)*P LIN+0.5)*0.566-1.0)*P LIN+SJ
C CØUNT NP PØSITIVE ZERØ CRØSSINGS AND THEN CØMPUTE ELAPSED TIME TT(I,L)
   50 DØ 53 ICRØSS=1,NP
   51 J=J+1
      IF(SPN(J)  )52,52,51
   52 J=J+1
      IF(SPN(J)  )52,53,53
   53 CØNTINUE
      SJ = J
      TT(I,L) = SJ-START
      IF(SPN(J)  )54,55,54
   54 P LIN=SPN(J)/(SPN(J)-SPN(J-1))
      TT(I,L)  =(((P LIN-1.5)*P LIN+0.5)*0.566-1.0)*P LIN+TT(I,L)
   55 CØNTINUE
C     MØDULØ 32 CØUNTER ØUTPUT WITH SLIDING PRESET
      DØ 64 JK=1,LSCALE
       ISET =0
       DØ 64 I=1,LØUT
C    CHANGE THIS FØR CHANGE IN LØUT
      IF (MØD(I,7)+0 .EQ. 0) ISET = ISET+1
64    IT(I,JK) = MØD(IDINT(TSCALE*TT(I,JK))+ISET,32)+0
      WRITE(3) IT
      WRITE(6,611) N
   60 CØNTINUE
      REWIND 1
      END FILE 3
      REWIND 3
      STØP
      END
```

CONFIDENTIAL

```
PRØGRAMS LØADED AND CØRE LIMITS
        MAINI              00600  -  01750
                           01751  -  47015
        E161               47016  -  47037
                           47040  -  47043
        E166               47044  -  47211
                           47212  -  47231
        DCØS               47232  -  47250
                           47251  -  47254
        DSIN               47255  -  47421
                           47422  -  47472
        DLØG               47473  -  47607
                           47610  -  47656
        DEXP               47657  -  50026
                           50027  -  50072
        DMØD               50073  -  50122
                           50123  -  50124
        DABS               50125  -  50141
        A162               50142  -  50156
                           50157  -  50160
        S162               50161  -  50200
                           50201  -  50202
        M162               50203  -  50217
                           50220  -  50221
        D162               50222  -  50242
                           50243  -  50246
        DBLE               50247  -  50260
        E121               50261  -  50277
                           50300  -  50301
        E122               50302  -  50412
                           50413  -  50425
        ALØG               50426  -  50523
                           50524  -  50537
        EXP                50540  -  50650
                           50651  -  50673
        MØD                50674  -  50720
                           50721  -  50721
      * SNGL               50722  -  50770
      * MC161              50771  -  51011
      * MC162              51012  -  51040
      * AS122              51041  -  51170
      * M122               51171  -  51262
      * D122               51263  -  51361
      * N122               51362  -  51366
      * C121               51367  -  51414
      * C171               51415  -  51423
      * C181               51424  -  51433
      * AINT               51434  -  51462
      * DINT               51463  -  51520
      * IAT                51521  -  51556
      * A171               51557  -  51574
      * C126               51575  -  51636
      * Z180               51637  -  51663
      * A181               51664  -  51714
      * AS166              51715  -  52143
      * M166               52144  -  52341
      * D166               52342  -  52547
      * N166               52550  -  52560
      * C162               52561  -  52614
      * C161               52615  -  52650
      * C116               52651  -  52676
      * C112               52677  -  52743
```

CONFIDENTIAL

```
* ZI70          52744  -  52761
* CIVT          52762  -  53035
* ARL1          53036  -  53073
* EXIT/IST      53074  -  53075
* FØRTI/9       53076  -  56616
* FIER          56617  -  56703
* IPA           56704  -  56721
* FØRT/MS-1     56722  -  57357
```

```
ENTRY PØINT AND CØMMØN BLØCK LØCATIØNS
                NAME               LØCATIØN
                FBLANK             00000
                MAIN1              00602
                IRE                56536
                IPA                56704
                IRD                53161
                I2I                55625
                INE                55704
                E122               50304
                C126               51575
                E121               50263
                E161               47020
                M166               52144
                D166               52342
                M162               50205
                C116               52651
                A166               51736
                D162               50224
                IWR                53166
                MØD                50676
                S166               51715
                DCØS               47234
                A162               50144
                C162               52561
                S122               51041
                D122               51263
                S162               50163
                IDINT              50737
                IIX                53652
                IND                53714
                IEN                56531
                IST                53074
                IAT                51521
                E166               47046
                DLØG               47475
                DEXP               47661
                DMØD               50075
                FIER               56617
                C161               52615
                DSIN               47257
                DABS               50127
                N166               52550
                C181               51424
                Z180               51637
                A181               51664
                DINT               51463
                DBLE               50251
                C112               52677
                ALØG               50430
                M122               51171
                EXP                50542
                AINT               51434
                C121               51367
                C171               51415
                Z170               52744
                A122               51047
                N122               51362
                A171               51557
                MC162              51012
                MC161              50771
                ERRØR              56631
```

```
ARLI                53036
CIVTARGLI           53020
ARGLI               53061
CIVTAQ              53000
CIVTQ               52762
IIFA                51533
MØNII               74506
IBA                 56524
IIF                 53646
IID                 53641
IIE                 53634
IDE                 53103
IEC                 53076
IEF                 53142
IØRETI              53341
IØTBLI              57212
WRITEI              56740
SETEØF              56722
READI               56735
BACKSPACEI          56727
ENDFILEI            56732
REWINDI             56743
```

PROGRAM LISTING:

Procs Subroutine

```
          JOB       D603      WPP PRGRM 04.00     JOSE M-F  SIMLTNTST   C0000100
          FORTRAN   LM                                                  C0000200
C$$$$ *****************************TWO OUTPUT TAPES WILLBE USED*********C0000300
      DIMENSION IT(11426),AIT(11426),ANUM(16424),TEMP(224),             C0000400
     DBUFFER(288),IBUFER(288),ANUM2F(224),ISTAT(113),CUM(113)           C0000500
      REAL ABUF(224),BBUF(224)                                          C0000600
      EQUIVALENCE(IT(1),AIT(1),ANUM(1)),(TEMP(1),ANUM2F(1))             C0000700
        DO 6000 JTEST=1,12,1                                            C0000800
C**********DATA  CARDS INSERT*****************                          C0000900
      REWIND 1                                                          C0001000
       REWIND 2                                                         C0001100
      REWIND 3                                                          C0001200
      NFILE=1211                                                        C0001300
 1000          FORMAT(I1)                                               C0001400
          READ 1000, IPEAK                                              C0001500
          READ1001,LSCALE                                               C0001600
 1001     FORMAT(I2)                                                    C0001700
          IF(LSCALE.EQ.0)GOTO120                                        C0001800
          JSCALE=LSCALE-1                                               C0001900
          NSIG=101*JSCALE+100                                           C0002000
          DO 0001J=1,NSIG
      0001   READ(3)(IBUFER(I),I=1,224)
      120    NRCDS=100
     C***************READ 11232 IT'S  FROM  TAPE UNIT 3*********
        IF(LSCALE.GT.0)NRCDS=101
          JRCDS=11200
          MRCDS=50
        DO 100 JJ=1,2,1
          J=224
        N=1
          DO 101 I=1,MRCDS,1
         READ(3)(IT(K),K=N,J)
          N=N+224
          J=J+224
      101     CONTINUE
        J=1
          DO 103 I=1,JRCDS,1
        CALL ZONEIT(IT(I),ABUF(J),BBUF(J))
        IF(J.NE.224)GOTO105
     C**********THE AIT'S WILL BE IN TAPE UNIT 1  ****BAIT'S TAPE 2****
```

```
      J=1                                                             C0004100
      WRITE(2)(BBUF(L),L=1,224)                                       C0004200
      WRITE(1)(ABUF(N),N=1,224)                                       C0004300
        GOTO 103                                                      C0004400
  105 J=J+1                                                           C0004500
  103   CONTINUE                                                      C0004600
          IF(NRCDS.EQ.101)MRCDS=51                                    C0004700
          IF(LSCALE.GT.0)JRCDS=11424                                  C0004800
  100    CONTINUE                                                     C0004900
      REWIND 1                                                        C0005000
C*********THE ABOVE STATMRNTS COMPUTE AIT'S***BAIT'S****              C0005100
C**** ****2ND PHASE COMPUTES A*****ANUMA2F*************************    C0005200
C*******READ 11424  AIT'S   FOR   PROCCESING*****************         C0005300
      ANUMA=0                                                         C0005400
      JTIME=0                                                         C0005500
      K=1                                                             C0005600
      DO 299 LOOP=1,2,1                                               C0005700
      IF(JTIME.EQ.0)GOTO210                                           C0005800
        REWIND 2                                                      C0005900
          J=224                                                       C0006000
          N=1                                                           C0006100
          DO 218 JJ=1,51,1                                              C0006200
          READ(2)(AIT(K),K=N,J)                                         C0006300
          N=N+224                                                       C0006400
  218     J=J+224                                                       C0006500
        GOTO220                                                         C0006600
  210     J=224                                                         C0006700
          N=1                                                           C0006800
          DO200 JJ=1,51,1                                               C0006900
          READ(1)(AIT(K),K=N,J)                                         C0007000
          N=N+224                                                       C0007100
  200     J=J+224                                                       C0007200
  220   DO 201 J=1,224                                                  C0007300
        ANUMA=ANUMA+AIT(J)                                              C0007400
  201   CONTINUE                                                        C0007500
        WOC=AIT(1)                                                      C0007600
        ANUMA2=ANUMA                                                    C0007700
        IF(ANUMA2.LT.112.0)ANUMA2=224.0-ANUMA2                          C0007800
        ANUM2F(1)=ANUMA2                                                C0007900
        N=2                                                             C0008000
```

```
      DO202  J=2,11201                                                   C0008100
      WOCNEW=AIT(J)                                                      C0008200
      ANUMA=ANUMA-WOC+AIT(J+223)                                         C0008300
      ANUMA2=ANUMA                                                       C0008400
      IF(ANUMA2.LT.112.0)ANUMA2=224.0-ANUMA2                             C0008500
      ANUM2F(N)=ANUMA2                                                   C0008600
      WOC=WOCNEW                                                         C0008700
      N=N+1                                                              C0008800
      IF(N.NE.225)GOTO202                                                C0008900
      N=1                                                                C0009000
      IF(JTIME.EQ.1)GOTO213                                              C0009100
      WRITE(2)(ANUM2F(L),L=1,224)                                        C0009200
      GOTO202                                                            C0009300
 213     WRITE(1)(ANUM2F(L),L=1,224)                                     C0009400
C*******THE ANUM2F WILL BE WRITTEN FOLLOWING THE BAITS OR BBUF******     C0009500
 202  CONTINUE                                                           C0009600
C*******TRANSFER REMAINIG AITS TO BEGINING*********                      C0009700
         MRCDS=49                                                        C0009800
         IF(LSCALE.GT.0)MRCDS=50                                         C0009900
      II=1                                                               C0010000
      DO203I=11201,11424                                           C0010100
      AIT(II)=AIT(I)                                               C0010200
  203 II=II+1                                                      C0010300
      KK=0                                                         C0010400
      IF(JTIME.EQ.0)GOTO214                                        C0010500
          J=448                                                    C0010600
          N=225                                                    C0010700
          DO215 JJ=1,MRCDS,1                                       C0010800
          READ(2)(AIT(K),K=N,J)                                    C0010900
          N=N+224                                                  C0011000
  215     J=J+224                                                  C0011100
      GOTO216                                                      C0011200
  214     J=448                                                    C0011300
          N=225                                                    C0011400
          DO 204 JJ=1,MRCDS,1                                      C0011500
          READ(1)(AIT(K),K=N,J)                                    C0011600
          N=N+224                                                  C0011700
  204     J=J+224                                                  C0011800
  216 WOC=AIT(1)                                                   C0011900
      N=1                                                          C0012000
```

```
            JRCDS=11202                                                 C0012100
            IF(LSCALE.GT.0)JRCDS=11426                                  C0012200
            DO 206 J=2,JRCDS,1                                          C0012300
      WOCNEW=AIT(J)                                                     C0012400
      ANUMA=ANUMA-WOC+AIT(J+223)                                        C0012500
      ANUMA2=ANUMA                                                      C0012600
      IF(ANUMA2.LT.112.0)ANUMA2=224.0-ANUMA2                            C0012700
      ANUM2F(N)=ANUMA2                                                  C0012800
      WOC=WOCNEW                                                        C0012900
      N=N+1                                                             C0013000
      IF(N.NE.225) GOTO206                                              C0013100
      N=1                                                               C0013200
C$$$$ THE ANUMA2'S WILL BE WRITTEN ON TAPE 2 FOLLOWING *AIT'S***********C0013300
      IF(JTIME.EQ.1)GOTO217                                             C0013400
      WRITE(2)(ANUM2F(L),L=1,224)                                       C0013500
      GOTO206                                                           C0013600
  217 WRITE(1)(ANUM2F(L),L=1,224)                                       C0013700
  206 CONTINUE                                                          C0013800
      JTIME=1                                                           C0013900
      ANUMA=0                                                           C0014000
```

```
 299   CONTINUE                                                        C0014100
C**********333THIRD PHASE****ANUMS **ARE   ****COMPUTED********************C0014200
         REWIND3                                                       C0014300
        DO 88 I=1,NFILE,1                                              C0014400
 88     READ(3)(IBUFER(K),K=1,224)                                     C0014500
       REWIND 1                                                        C0014600
       REWIND 2                                                        C0014700
          DO 030 I=1,NRCDS                                             C0014800
      READ(1)(BUFFER(L),L=1,224)                                       C0014900
 030  READ(2)(BUFFER(L),L=1,224)                                       C0015000
      JU=50                                                            C0015100
 306  DO 300 KKK=1,2,1                                                 C0015200
      N=1                                                              C0015300
      DO301  LL=1,JU,1                                                 C0015400
      READ(1) (ABUF(L),L=1,224)                                        C0015500
      READ(2) (BBUF(J), J=1,224)                                       C0015600
      DO 302 L=1,224,1                                                 C0015700
      ANUM(N)=AMAX1(ABUF(L),BBUF(L))                                   C0015800
 302  N=N+1                                                            C0015900
 301   CONTINUE                                                        C0016000
```

```
          JRCDS=11200                                                   C0016100
          IF (LSCLE.GT.0)JU=51                                          C0016200
      N=1                                                               C0016300
          WRITE(3)(ANUM(LL),LL=1,JRCDS)                                 C0016400
          IF(LSCALE.GT.0)JRCDS=11424                                    C0016500
 300       CONTINUE                                                     C0016600
C******FOURTH PHASE ***CUMULATIVE DISTRIBUTION IS COMPUTED************* C0016700
C********INSERT INSTRUCTION FOR ERROR ******* IF ANUM IS GREATER THAN 22C0016800
          DO 400 I=1,113                                                C0016900
      CUM(I)=0                                                          C0017000
 400     ISTAT(I)=0                                                     C0017100
          GO TO (4001,4002,4003,4004),IPEAK                             C0017200
 4001     NFIN=11200                                                    C0017300
          NDELTA=1                                                      C0017400
        TOTSPL=22400.00                                                 C0017500
          GOTO4000                                                      C0017600
 4002     NFIN=11200                                                    C0017700
          NDELTA=224                                                    C0017800
          TOTSPL=100.0                                                  C0017900
          GOTO4000                                                      C0018000
```

```
4003      NFIN=33                                                  C0018100
          NDELTA=1                                                 C0018200
           TOTSPL=33.0                                             C0018300
          DO 4031 KX=1,224                                         C0018400
4031      BUFFER(KX)=ANUM(KX)                                      C0018500
          DO 4032 KX=1,10976                                       C0018600
4032      ANUM(KX)=ANUM(KX+224)                                    C0018700
          DO 4033 I=1,16,1                                         C0018800
          IXX=672*I                                                C0018900
          IX=IXX-670                                               C0019000
          ANUM(I)=ANUM(IX-1)                                       C0019100
          DO 4033 J=IX,IXX                                         C0019200
4033      ANUM(I)=AMAX1(ANUM(I),ANUM(J))                           C0019300
          DO 4034KX=1,16                                           C0019400
4034      TEMP(KX)=ANUM(KX)                                        C0019500
          REWIND 3                                                 C0019600
          DO 005 IK=1,NFILE                                        C0019700
005       READ(3)(IBUFER(L),L=1,224)                               C0019800
          READ(3)(ANUM(L),L=1,11200)                               C0019900
          KXX=1                                                    C0020000
          DO 4035 KX=11201,11424                                        C0020100
          ANUM(KX)=BUFFER(KXX)                                          C0020200
 4035     KXX=KXX+1                                                     C0020300
          DO 4036 I=1,17,1                                              C0020400
          IXX=672*I                                                     C0020500
          IX=IXX-670                                                    C0020600
          ANUM(I)=ANUM(IX-1)                                            C0020700
          DO 4036J=IX,IXX                                               C0020800
 4036     ANUM(I)=AMAX1(ANUM(I),ANUM(J))                                C0020900
          DO 4037KX=18,33                                               C0021000
 4037     ANUM(KX)=TEMP(KX-17)                                          C0021100
          DO 4038JK=1,33,1                                              C0021200
          INDEX=ANUM(JK)-111.0                                          C0021300
 4038     ISTAT(INDEX)=ISTAT(INDEX)+1                                   C0021400
          GO TO 410                                                     C0021500
C**************FOR FOURTH DETECTION************************************ C0021600
 4004     CONTINUE                                                      C0021700
 4000     CONTINUE                                                      C0021800
          IF(LSCALE.GT.1)NFIN=11424                                     C0021900
          DO 403JK=1,NFIN,NDELTA                                        C0022000
```

[redacted]

```
          INDEX=ANUM(JK)-111.0                                  C0022100
   403     ISTAT(INDEX)=ISTAT(INDEX)+1                          C0022200
          REWIND 3                                              C0022300
             DO 004 IK=1,NFILE,1                                C0022400
   004         READ(3)(BUFFER(KO),KO=1,224)                     C0022500
          READ(3)(ANUM(L),L=1,11200)                            C0022600
             DO 404JK=1,11200,NDELTA                            C0022700
          INDEX=ANUM(JK)-111.0                                  C0022800
   404     ISTAT(INDEX)=ISTAT(INDEX)+1                          C0022900
   410          CUM(113)=ISTAT(113)                             C0023000
          JAA=0                                                 C0023100
          DO 406 I=1,112,1                                      C0023200
            JA=113-I                                            C0023300
          STAT=ISTAT(JA)                                        C0023400
           JAA=JA+1                                             C0023500
   406       CUM(JA)=CUM(JAA)+STAT                              C0023600
           DO405 I=1,113                                        C0023700
   405      CUM(I)=CUM(I)/TOTSPL                                C0023800
            PRINT25,(ISTAT(JOE),JOE=1,113)                      C0023900
   25     FORMAT(1I10)                                          C0024000
```

[redacted]

```
      PRINT26,(CUM(I),I=1,113)                                          C0024100
26    FORMAT(1F7.5)                                                     C0024200
6000      PRINT6001,IPEAK,LSCALE                                        C0024300
6001      FORMAT(I3,I6)                                                 C0024400
      STOP                                                              C0024500
      END                                                               C0024600
                                                                        C0024700
      SUBROUTINE ZONEIT(IT,ABUF,BBUF)                                   C0024800
      DIMENSION ZONA(8),ZONB(8)                                         C0024900
      INTEGER SUMZA,ZONA,SUMZB,ZONB                                     C0025000
      SUMZA=0                                                           C0025100
      SUMZB=0                                                           C0025200
      DO 1 K=1,8                                                        C0025300
      M=K-1                                                             C0025400
      IF(IT.GE.M)GOTO2                                                  C0025500
      GOTO51                                                            C0025600
2     IF(IT.LE.15+M)GOTO3                                               C0025700
51    ZONA(K)=0                                                         C0025800
      GO TO 4                                                           C0025900
    3 ZONA(K)=+1                                                        C0026000
    4 SUMZA=SUMZA+ZONA(K)                                          C0026100
      IF(IT.GE.8+M)GOTO11                                          C0026200
      GOTO52                                                       C0026300
   11 IF(IT.LE.23+M)GOTO5                                          C0026400
 52   ZONB(K)=0                                                    C0026500
      GO TO 6                                                      C0026600
    5 ZONB(K)=+1                                                   C0026700
    6 SUMZB=SUMZB+ZONB(K)                                          C0026800
    1 CONTINUE                                                     C0026900
      TSUMB=FLOAT(SUMZB)/8.0                                       C0027000
      TSUMA=FLOAT(SUMZA)/8.0                                       C0027100
      ABUF=TSUMA                                                   C0027200
      BBUF=TSUMB                                                   C0027300
      RETURN                                                       C0027400
      END                                                          C0027500
$END                                                               C0027600
$DATA                                                              C0027700
3                                                                  C0027800
 0                                                                 C0027900
```